

počítačová elektronika

HARDWARE * SOFTWARE * INFORMACE



... A OPĚT ZELENÁ ROČENKA

Je to známkou toho, že už zase uplynul rok. Zelená ročenka přichází tentokrát do zcela jiného "prostředí" - do konkurence obrovského množství nejrozličnějších časopisů a publikací všeho druhu, do doby, kdy všichni obrátí každý peníz třikrát, než ho vydají. Její cena je solidní - je levnější, než Amatérské radio, má dvojnásobný rozsah za jedenapůlnásobnou cenu. I její obsah je solidní. Jsou zde opět nejlepší příspěvky z Mikrokonkursu AR, u některých je i celý výpis programu - ne snad kvůli okopírování, ale jako typický příklad určitého řešení, z kterého se lze poučit a čerpat při řešení podobných problémů v jiných aplikacích.

Jako předzvěst rozšiřování osobních počítačů i do "amatérského" světa jsme přidali dva rozsáhlé materiály: Informaci o volně šířených programech s možností objednání, a Informace o použití personálních počítačů k měření a řízení.

OBSAH:

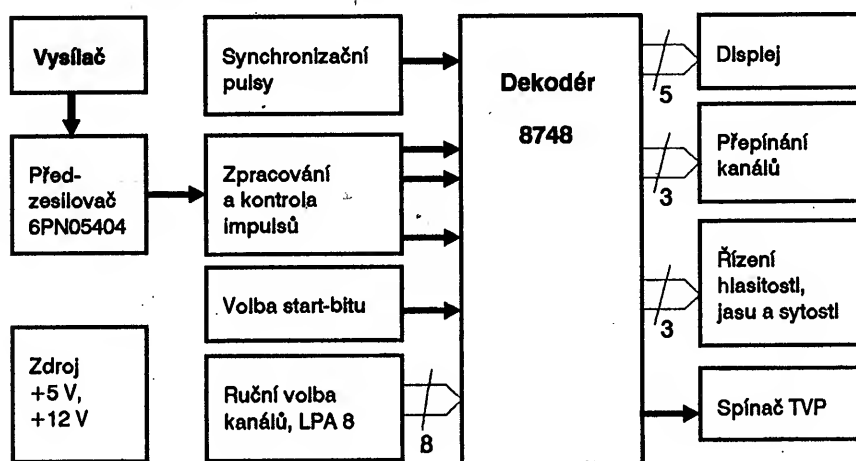
Dálkové ovládání televizoru	2
Automatizace průmyslových a laboratorních procesů pomocí personálních počítačů	26
PUBLIC DOMAIN - volně šířené programy	33
* úvod	
* seznam	
* popis	
<i>(jako vyjímání 16ti stránkový přehled)</i>	
FILE OUT & FILE IN	49
Simulátor jednočipových mikropočítačů	57
Emulátor pro jednočipové mikropočítače na ZX-Spectru	74
Logický analyzátor z ATARI 800 XE/XL	77
Počítače TANDON	III. a IV. str. obálky

Vydal Magnet-Press Praha. Index 46043. Adresa redakce: Jungmannova 24, 113 66 Praha 1, tel. 260651-7.
Šéfredaktor Ing. Jan Klíbal, odpovědný redaktor Ing. Alek Myslík. Sekretářka redakce T. Trnková, linka 355.
Výtisklo Naše vojsko, závod 02, Vlastina 889/23, Praha 6. Za původnost a správnost příspěvku ručí autor.

DÁLKOVÉ OVLÁDÁNÍ TELEVIZORU

Ing. Eduard Hoffmann, Strážnická 14, 627 00 Brno

Když jsem po přečtení [1] a prostudování [2] zjistil, že dálkové ovládání nejsou žádné čáry, začal jsem uvažovat o jeho zabudování do BTVP COLOR 419. Trochu mi však vadilo, že u nás používané obvody dálkového ovládání (dále jen DO) neumožňují zachovat ruční volbu programů osmi tlačítky, ale dovolují jen dvě tlačítka PROGRAM+ a PROGRAM-, což by změnilo vzhled TVP a znesnadnilo ruční obsluhu (řešení [1] nahrazuje dokonce původní přepínací jednotku jediným tlačítkem). Navíc obvod U806D, používaný pro DO v našich televizorech, se v prodejnách TESLA téměř nevyskytuje. Poté, co jsem získal příležitost koupit mikropočítač 8748 i s krystalem za částku, srovnatelnou s cenou obvodu U806D, rozhodl jsem se postavit dálkové ovládání zcela nové.



Obr. 1. Blokové schéma

Ústředním prvkem této konstrukce je mikropočítač typu 8748, který je obklopen obvody, jejichž skladba je dána obsahem mého šuplíku. Vazba na televizní obvody byla převzata z přijímačů COLOR 425 a podobných. Hlavní rysy mého řešení jsou tyto:

- ovládání základních funkcí je shodné s továrními výrobky (používám vysílač, který se dodává např. k TVP COLOR 425),
- pamatuje se hlasitost, jas a sytost pro každý kanál; tyto hodnoty se automaticky nastaví při přepnutí kanálu a není tedy nutné obraz a zvuk při přepínání doladovat. Vnitřní paměť mikropočítače lze navíc zálohovat z akumulátoru a pamatovat si tak nastavené hodnoty hlasitosti, jasu a sytosti pro všechny kanály i při vypnutí TVP ze sítě,
- zvolený kanál se indikuje na sedmi-segmentovém zobrazovači,
- po vypnutí TVP dálkovým ovladačem (režim POKOTOVOST) lze naprogramovat čas, po jehož uplynutí dojde k opětovnému zapnutí TVP a přepnutí na kanál, který byl zvolen před vypnutím. Je možno naprogra-

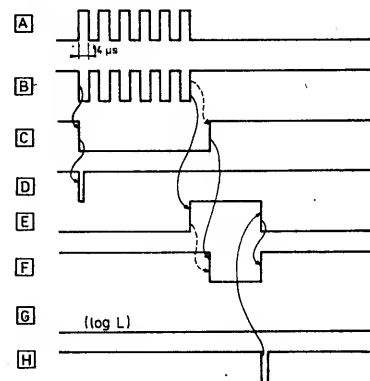
movat čas nového zapnutí TVP za 1 až 16383 minut (cca 11 dnů),

- tlačítko, které u továrních výrobků slouží k zobrazení čísla zvoleného kanálu v rohu obrazovky, dostalo funkci „co je kde na programu?“. Při jeho stisku se s periodou asi 5 s postupně zapnou všechny obsazené kanály,
- 5 minut po skončení vysílání se TVP automaticky vypne,
- zůstala zachována upravená jednotka předvolby, lze tedy ručně přepnout na libovolný kanál přímo,
- dálkově nelze ovládat AFC.

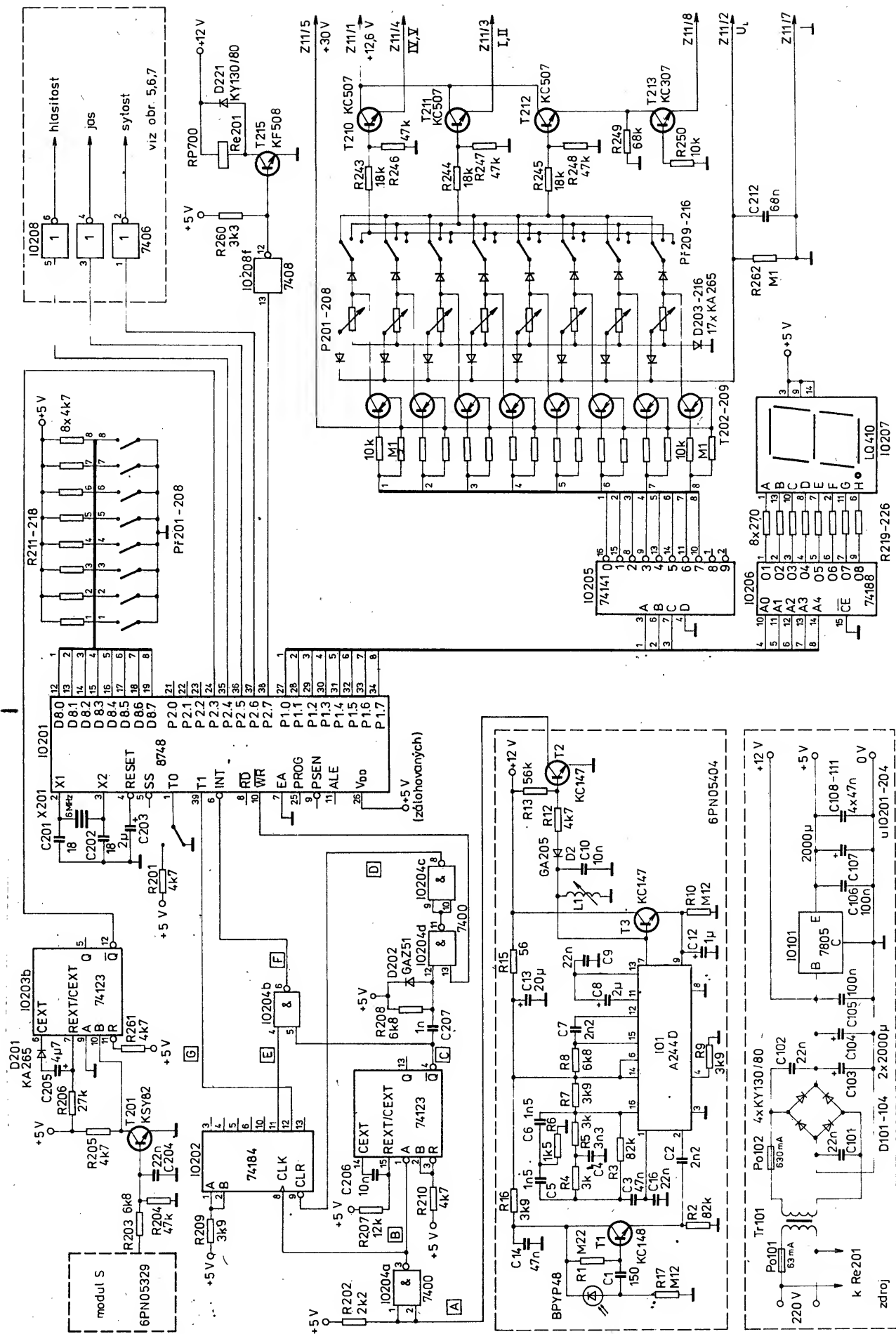
Popis hardware

Blokové schéma je na obr. 1, podrobné schéma zapojení je na obr. 3. Předzesilovač DO (6PN05404) je někdy k dostání v prodejnách TESLA za 135 Kčs nebo se dá postavit podle obr. 2 nebo podle [3] (místo fotodiody BPYP46 používám paralelně spojené fototranzistory KPX81 za 4,80 Kčs). Vysílač DO lze koupit za 570 Kčs nebo si jej postavit např. podle [2], [3].

Z předzesilovače přichází sériově vysílaná informace ve formě skupin šesti impulsů, které jsou detekovány obvody IO202, IO203 a IO204. Činnost obvodů je zřejmá z obr. 2, podotýkám jen, že do mikropočítače se nedostanou rušivé signály, které mají méně než šest impulsů, pokud by přišlo impulsů více, má o tom mikropočítač informaci na vstupu T1 (IO201, vývod 39) a takový signál odmítne. Vstupem T0 (IO201 vývod 1) je nastavena úroveň start-bitu, prodávané vysílače jsou zapojeny tak, že instrukce jsou vysílány s nulovým start-bitem (vývod T0 je tedy nutno uzemnit). Na vstupu P2.3 (IO201/24) se sleduje přítomnost synchronizačních pulsů v televizoru, na výstupech P2.4, P2.5 a P2.6 (IO201 - 35, 36 a 37) jsou generovány šifrové modulované pulsy, které po zintegrování řídí hlasitost, jas a sytost v 64 stupních, výstup P2.7 (IO201/38) ovládá relé pro zapínání TVP. K bráně DB je připojeno osm tlačítek pro ruční volbu programů, z brány P1 se řídí přepínání kanálů a zobrazovač. Místo dekodéru 74188 lze použít obvody D147C,D nebo D146C,D a tečku na zobrazovači spínat jedním zbývajícím invertorem 7406 z výstupu P1.7 (IO201/34).



Obr. 2. Průběh signálů A - H z obr.3



Obr. 3. Celkové schéma zapojení dálkového ovládání televizoru

Popis software

Řídící program je připraven na možnost zálohování vnitřní paměti RAM mikropočítače z baterie nebo akumulátoru na vývodu IO201/26, přestože u nás vyráběné MHB8748C mají poměrně velký odběr proudu. Po zapnutí se provede test, zda paměť RAM je zálohovaná (řádky 334 - 350), pak proběhne jednoduchý test mikropočítače a důkladný test paměti EPROM (řádky 986 až 1080); při jejich provádění se na zobrazovači pro kontrolu ukazují znaky A (při testu mikropočítače) a B, C, D, E (při testu paměti EPROM). Pak se inicializují hodnoty některých proměnných (ř. 352 - 386) a zapne se televizor a poslední sledovaný kanál v případě, že paměť je zálohovaná a TVP byl vypnut síťovým vypínačem. V ostatních případech (paměť není zálohovaná nebo paměť je zálohovaná, ale TVP byl vypnut DO) se přejde do pohotovostního stavu I; do provozního stavu se TVP dostane ruční nebo dálkovou volbou programu. Program pak čeká na příchod instrukce (ř. 693 - 833) a po jejím přijetí provede kontrolu start-bitu (ř. 411 - 417) a případné zpracování (ř. 433 - 650). Během čekání na instrukci se testují tlačítka a při jejich změně se přepne kanál podle nového stavu tlačítek (ř. 741 - 760). Program je periodicky přerušován vnitřním čítačem mikropočítače, při obsluze přerušeni (ř. 137 - 305) se měří čas a jsou generovány šifrově modulované impulsy pro ovládání hlasitosti, jsou a syst.

Obsluha

Jak již bylo řečeno, obsluha základních funkcí DO se shoduje s továrními výrobky a ruční ovládání zůstalo zachováno (navenek se TVP od původního přístroje odlišuje jen sedmissegmentovým zobrazovačem).

Funkce, které jsou navíc, si teď popíšeme podrobněji:

1. Při stisku tlačítka, které u továrních TVP zobrazí v levém horním rohu obrazovky číslo kanálu, se postupně zapnou s časovým odstupem asi 5 s (nebo 1,5 s když kanál nevysílá) všechny obsazené kanály (v našem případě kanály 1 až 5), „prohlížení“ programů se zastaví při návratu na výchozí kanál opětovným stiskem tlačítka, případně tlačítka 1 až 8 na dálkovém ovladači nebo na televizoru. Je-li např. zvolen kanál 2, zapnou se postupně kanály 3-4-5-1-2, při sledování kanálu 7 se zapnou kanály 1-2-3-4-5-7.

2. Po vypnutí TVP pomocí DO je možné dálkovým ovladačem naprogramovat čas, po jehož uplynutí dojde opět k zapnutí. K popisu této nestandardní funkce je nutné předdeslat, že celé zařízení se může nacházet v několika stavech:

a) PROVOZ

- TVP je zapnut, na zobrazovači je číslo zvoleného kanálu.

b) POHOTOVOST I

- TVP je vypnut, na zobrazovači bliká pomalu tečka (není naprogramován čas).

c) PROGRAMOVÁNÍ

- TVP je vypnut, na zobrazovači svítí vkládaná číselnice programovaného času.

d) POHOTOVOST II

- TVP je vypnut, na zobrazovači bliká rychle tečka (je naprogramován čas).

K programování času se využívají tlačítka AFC+ (zahájí a ukončí programování) a AFC- (zruší naprogramovaný čas a převede TVP zpět do režimu POHOTOVOST I). Jednotlivé číslice se zobrazují na displeji a zadávají se tlačítky 1 až 8, sousední tlačítka \rightarrow a \leftarrow slouží k zadání číslic 9 a 0; horní dvě řady tlačítek na vyslači, které mají v provozním režimu význam

1 2 3 4	\square
5 6 7 8	\rightarrow

fungují tedy při programování času jako

1 2 3 4 0
5 6 7 8 9

ostatní tlačítka (kromě AFC+ a AFC-) jsou nefunkční. Lze naprogramovat čas 1 minuta až 16383 minut, při překročení tohoto rozsahu je programovací režim automaticky zrušen, při vložení každé číslice se pro kontrolu, že číslice byla přijata, změni tečka na zobrazovači. Pro názornost si ukažme např. naprogramování času 220 minut:

krok	stisk	displej	význam
1	I	pomalá tečka	vypnutí TVP, režim POHOTOVOST I
2	AFC+	.0	přechod do režimu PROGRAMOVÁNÍ
3	2	2	vložení cifry 2
4	2	.2	vložení cifry 2
5	\square	0	vložení cifry 0
6	AFC+	rychlá tečka	přechod do režimu POHOTOVOST II, od tohoto okamžiku se měří čas

Po uplynutí 3 hodin a 40 minut dojde k zapnutí TVP.

3. Mikropočítač neustále sleduje přítomnost synchronizačních pulsů v TVP. Při skončení vysílání (pulsy přestanou přicházet) se stáhne hlasitost, jas a sytost na minimum a začne se odměřovat čas. Pokud se do 5 minut neobnoví vysílání nebo nedojde k přepnutí na vysílající kanál, TVP se vypne (přejde do režimu POHOTOVOST I).

Oživení

Oživení a nastavení předzesilovače DO je dobře popsáno v [4]. Zbytek je natolik jednoduchý, že se dá bez problémů oživit jen s pomocí Avometu a logické sondy. Postup je následující:

1. zapájet všechny součástky, mikropočítač neosazovat,

2. ohmmetrem zkontrolovat, zda není zkratováno napájení,

3. oživit zdroj, připojit +5 V a +12 V a zkontrolovat napájecí napětí všech integrovaných obvodů (na zobrazovači svítí trvale tečka),

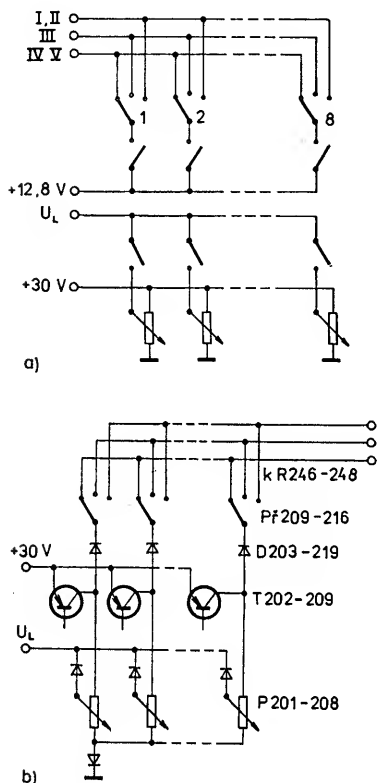
4. kusem drátu postupně uzemnit vstupy IO206 (částečná prověrka zobrazovače), odpory R202 a R205 (logickou sondou ověřit, že kývají obě poloviny IO203), odpory R211 až R218 (zkontrolovat cestu k příslušným vývodům mikropočítače), vstup IO208/13 (spíná relé) a vstupy 1, 3, 5 obvodu IO208 (voltmetrem měřit nabíjení a vybíjení kondenzátorů C208, C210, C211),

5. připojit oživený předzesilovač DO a při vysílání zkontrolovat logickou sondou průchod signálu jednotlivými obvody až k patici IO201. Je-li předzesilovač správně nastaven, musí posuvný registr napočítat vždy šest impulsů; to znamená, že na výstupech QA až QF tohoto registru (IO202/3-6, 10, 11) logická sonda při stisku tlačítka na vyslači DO bliká a po uvolnění tlačítka zůstane ve stavu H, výstupy QG a QH (IO202/12, 13) jsou trvale ve stavu L. Pokud tomu tak není, musíme osciloskopem změřit délku kyvu IO203A (nesmí být kratší než 28 mikrosekund) a šířku impulsu na vstupu IO202/9 (musí být kratší než 14 mikrosekund),

6. zasunout mikropočítač do patice; pokud je na jeho vývodech 26 a 40 napětí +5 V a mezi vývody 2 a 3 je zapojen rezonanční obvod, musí mikropočítač pracovat (kmitá vývod ALE). Je-li paměť EPROM správně naprogramována, proběhnou na zobrazovači rychle za sebou znaky A, B, C, D, E a pak začne pomalu blikat tečka; pokud tomu tak není, je procesor vadný nebo obsah EPROM není správný. V této souvislosti musím ještě upozornit na článek [5], podle kterého nepatří obvody MHB8748C k nejspolehlivějším; já používám obvod firmy NEC bez jakýchkoliv problémů, ze dvou zkoušených obvodů MHB byl jen jeden schopen pracovat správně déle než 5 minut. Výstup ALE (IO201/11) spojíme s kolektorem tranzistoru T201, bude dočasně zastávat funkci zdroje synchronizačních pulsů (při jejich nepřítomnosti svítí na zobrazovači tečka a nejde měnit hlasitost, jas a sytost),

7. kusem uzemněného drátu slmulíme tlačítka, musí sepnout relé a na zobrazovači se ukázat odpovídající číslice. Připojíme předzesilovač DO

a zkusíme přepínat kanály vysílačem DO, v této fázi už musí mikropočítač správně reagovat. V případě, že tomu tak není, zkontrolujeme ještě jednou vývody IO202 podle bodu 5 při zasunutém mikropočítači, nedostatky jde ještě „doladit“ tak, že vstupy 6 a 39 IO201 připojíme na ty výstupy IO202, které pracují způsobem, popsáním v bodě 5. Pokud mikropočítač na vysílané instrukce reaguje, je číslíková část zařízení oživena.



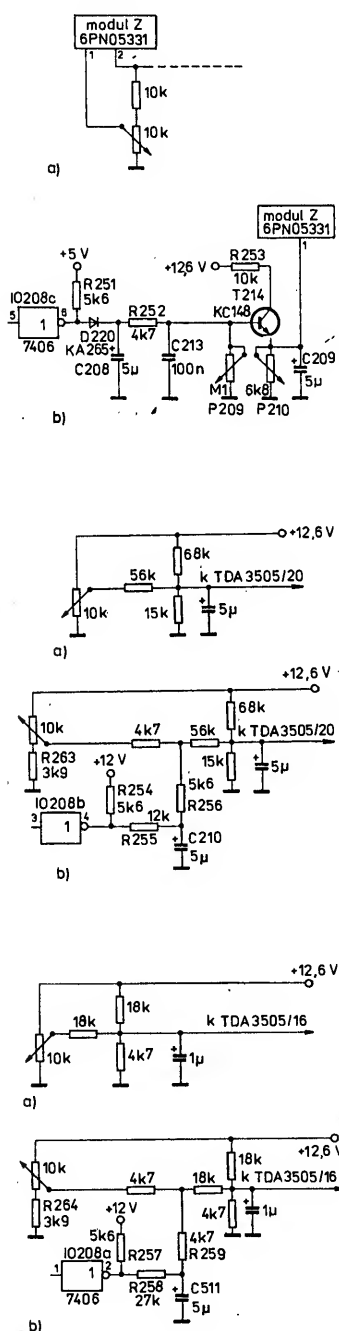
Obr. 4. Úprava jednotky předvolby

Praktická realizace

Zapojení bylo realizováno drátovým spoji na kusu univerzální desky. V televizoru je třeba provést některé zásahy, z nichž nejpracnější je úprava jednotky předvolby LPA8 (obr. 4). Tu vyjeme, rozebereme a upravíme: přeškrábeme odporové dráhy, zanýtujeme nýty 1,6 x 2,5 mm pro pájení vodičů, vyjeme hřebínek, který přivádí 12,6 V na přepínač pásem a ohneme přívody jezdců od ladicích potenciometrů. Po sestavení jednotky jsem připájel katody diod D212 až D219 přímo na očka, která tvoří kontakt s třípolohovým přepínačem pásem. Otvory, kudy vedl přívod k jezdcům ladicích potenciometrů, jsem protáhl úzké pásky pocínovaného plechu, které slouží jako kontakty nových přepínačů. K nim jsou připájeny přívody od brány DB mikropočítače.

Umístění zobrazovače, fotodiody a desky s plošnými spoji si nejlépe každý zvolí sám, připomínám jen, že

fotoprvek je nutné dobře odstínit před vnějším světlem (vhodný je kousek vyvolaného neosvětleného barevného pozitivního filmu AGFA). Po výměně potenciometru pro řízení hlasitosti upravíme podle obr. 5 obvody pro ovládání hlasitosti, jasu a sytosti; v části a) je vždy původní zapojení, v části b) je upravené zapojení. Při vyjmutém mikropočítači uzemníme vývod IO208 č. 5, potenciometr hlasitosti nastavíme na maximum a na trimru P210 nastavíme napětí 2,3 V; toto je jediná nastavovací operace. Nakonec ještě spojíme odpor R203 a špičku 3 modulu S (6PN 05329) a zkontrolujeme, zda je na vstupu P2.3 (IO201/24) úroveň H při odpojení a úroveň L při připojení anténního přívodu.



Obr. 5. Úpravy obvodů pro ovládání hlasitosti, jasu a sytosti

Závěr

Na závěr chci znovu zdůraznit, že díky použitému mikropočítači je celé zařízení velmi jednoduché a snadné je i jeho obsluha, i když u poloviny tlačítek došlo ke kumulaci funkcí (tři nestandardní byly ukázány). Obvod 8748 má kromě toho velké množství I/O linek a toto zapojení lze snadno modifikovat a použít je při zachování jádra programu a doplnění příslušných obslužných částí i k dálkovému ovládání jiných zařízení (u aplikací, kde by nezáleželo na přesném měření času by bylo dokonce možné zjednodušit celé zapojení výpuštěním obvodů IO202, IO203 a IO204 - pak by se ovšem řídicí program musel změnit od základu). Naopak U806D je jednoúčelový obvod, který k ovládání jiných zařízení využijeme stěží. Příkladám proto komentovaný výpis řídicího programu (zabírá celou vnitřní paměť EPROM mikropočítače) a pro úplnost i tentýž program v hexadecimální formě. Upozornuji ještě na skutečnost, že programová paměť je zabezpečena CRC s generujícím polynomem $x^{16} + x^{15} + x^2 + 1$ a pokud chceme tento vysoce účinný test paměti programu zachovat, musí každou změnu v řídicím programu provázet i změna obsahu na posledních dvou adresách (návod na výpočet CRC je uveden ve výpisu programu na řádcích 1022 - 1034).

Literatura

- [1] Tichý I., Černý P.: Dálkové ovládání k TVP ORAVAN. Konstrukční příloha časopisu AR 1988.
- [2] Teska V.: Dálkové ovládání výrobků spotřební elektroniky. AR B6/1987.
- [3] Teska V.: Dálkové ovládání výrobků spotřební elektroniky. AR B2/1988.
- [4] Žebrák M.: Obvody dálkového ovládání; AR B6/1987
- [5] Buchta J.: Zkušenosti s jednocipovými mikropočítači TESLA MHB 8748C. Sdělovací technika 12/1989.
- [6] Žebrák M.: Přijímače pro příjem barevné televize. Signálová část přijímačů. AR B4/1987, B5/1987.
- [7] Nedvěd J.: Přijímače pro příjem barevné televize. Rozkladové obvody. AR B5/1987, AR B6/1987
- [8] Schémata k TVP COLOR 419.
- [9] Schémata k TVP COLOR 425.

Seznam součástek:

IO1 A244D
IO101 MA7805
IO201 MHB8748C
IO202 MH74164
IO203 UCY74123
IO204 MH7400
IO205 MH74141
IO206 MH74188
IO207 LQ410
IO208 UCY7406N

D1 BPYP48
D2 GA205
D101-104, D221 KY130/80
D201, D203-220 KA265
D202 GA251

T1 KC149
T2,3 KC147
T201 KSY62
T202-209, T213 KC307
T210-212 KC507
T214 KC148
T215 KF508

P209 (TP160B/G) 100 k
P210 6k8 trimr

R1 M22
R2,3 82k
R4,5 3k0
R6 1k5
R7,9,16 3k9
R8 6k8
R10,17 M12
R12 4k7
R13 56k
R15 56
R201,205,210,211-218 4k7
R252,259,261 4k7
R202 2k2
R203,208 6k8
R204,246-248 47k
R206,258 27k
R207,256 12k
R209,263,264 3k9
R219,226 270
R227-234,250,253 10k
R235-242,262 M1
R243-245 18k
R249 68k
R251,254,256,257 5k6
R260 3k3

C1 150 ker.
C2,7 2n2 ker.
C3 47n ker.
C4 3n3 TGL5155
C5,6 1n5 TGL5155
C8 2μ ellyt
C9,16 22n ker.
C10 10n TC215
C13 20μ ellyt
C14 47n ker.
C101,102 22n ker.
C103,104,107 2m ellyt
C105,106,213 100n ker.
C108-111 47n ker.
C201,202 18 ker.
C203 2μ ellyt
C204 22n ker.
C205 4μ7 ellyt
C206 10n ker.
C207 1n ker.
C208-211 5μ ellyt
C212 68n ker.

X201 krystal 6000 kHz
RE201 relé, RP700
Po101 pojistka 63 mA
Po102 pojistka 630 mA

```
0000: C5 64 75 14 7F 04 00-D5 AF 16 0B 23 FD 62 76 6F
0010: C5 FF D5 72 5B 1B FB-53 3F 96 35 F1 AA 19 F1 AB
0020: 19 F1 AC B6 2D FA C6-30 9A EF CA 04 41 00 04 30
0030: 8A 10 00 04 41 F8 A9-B6 3D 99 7F 04 23 89 80 04
0040: 23 FB C6 49 9A DF CB-04 4E 8A 20 00 04 4E FC C6
0050: 56 9A BF CC 04 6F 8A-40 00 04 6F FA CA 96 6F FB
0060: CB 96 6F FC CC 96 6F-C5 BF 01 8A FF 89 78 B5 C5
0070: 1A 1B FB 96 7B 1C CD-23 10 4F AF 76 80 D5 FF 93
0080: FF B2 7D D2 8A FC 92-B5 04 B1 D5 19 F9 D3 FA 96
0090: AD A9 1A FA D3 FA 96-AD AA CC FC 96 AD FB CB 96
00A0: AD C5 AF A5 9A 7F B8-38 F0 54 F4 04 7D C5 FC 52
00B0: B5 89 80 04 7D 99 7F-04 7D 35 4F 04 BF 35 5F AF
00C0: 25 16 07 83 A3 83 B8-3A B9 FB BA 05 85 95 F9 E3
00D0: 20 D0 C6 D5 85 18 19-EA CE B8 1D 08 A0 B6 BE 74
00E0: 05 B8 38 A0 B8 20 B9-08 54 1C B9 B8 27 AF B8 20
00F0: B9 18 F0 03 BF E6 F9-B0 20 18 B9 F2 B8 38 F0 53
0100: 07 A0 8A FF A5 95 B6-11 FF 12 11 F0 54 F4 9A 7F
0110: B5 B5 85 FF 53 01 AF-25 55 54 26 97 67 26 20 A7
0120: F6 19 B9 3F 21 D1 96-19 F1 03 D0 F1 F6 19 34 38
0130: 23 02 14 B9 BD 04 24-19 03 CE E3 A8 76 40 24 45
0140: 03 F2 F6 DF F8 C8 03-49 B3 DF 6B F5 F5 F5 F5 F5
0150: F5 F5 F5 8F E2 E4 E8-5E 9A A1 B3 B9 BF D1 A5 B5
0160: 85 35 8A FF 89 78 23-01 14 BF 83 FF 76 E6 32 DF
0170: 52 86 23 04 14 B9 27-AC B8 38 F0 03 FC 27 F6 82
0180: F0 17 18 A0 44 F4 B8-39 F0 C8 A0 23 FB 04 BD FF
0190: 76 E6 72 DF B8 18 F0-A8 44 1E B8 18 F0 B6 A8 24
01A0: C4 B8 18 F0 B6 A8 24-D6 A8 FF 72 DF 27 A0 B8 1A
01B0: A0 85 83 B8 18 F0 17-24 C4 B8 18 F0 17 24 D6 B8
01C0: 18 F0 17 17 A8 FF 72-DF F0 17 D2 CE A0 83 B0 40
01D0: 83 B8 18 F0 17 17 A8-FF 72 DF F0 C6 DF 07 A0 83
01E0: A3 83 64 49 64 3F 64-12 FF 32 DF 72 DF 95 52 DF
01F0: B8 38 F0 24 FF FF 32-DF B2 E6 C8 23 38 28 A0 54
0200: F4 A5 9A 7F 23 9A 14-BD 83 2C 6A 2C 2B 79 2B 83
0210: 54 14 F6 1B 97 2C F7-2C 2B F7 2B 83 B0 13 18 B0
0220: 20 18 B0 20 18 83 86-A2 FD 96 2F 23 FD 14 BD FF
0230: 92 63 52 36 44 26 FC-03 AE F6 44 FF 37 72 26 FC
0240: 03 E5 B6 26 AC B8 39-10 F0 03 FB F0 E6 58 C8 F0
0250: 03 FB 27 E6 56 F0 18-A0 54 F4 B8 39 F0 C8 D0 96
0260: 26 44 7B 23 EF 14 BD-B8 1D 08 20 D0 C6 81 F0 74
0270: 05 72 26 B8 38 A0 54-F4 9A 7F A5 23 9A 14 BD 44
0280: 26 76 26 A0 72 8C 23-F7 14 BD 44 26 89 80 FF 72
0290: 26 65 43 08 AF 8A 70-D5 BA 00 BB 13 BC 13 C5 55
02A0: 44 26 46 A7 90 44 26-90 27 AE AA B9 01 86 B6 FA
02B0: 03 D5 B6 AD 44 26 46-BB 90 44 26 90 FA 03 F6 B6
02C0: 26 27 2A 03 E7 E6 CA-F9 4E AE F9 69 A9 F2 D7 44
02D0: AD 56 D6 90 44 26 90-86 D1 FA 03 D5 E6 D7 FE 83
02E0: F9 03 B4 B3 B8 EB EE-F1 FE 04 C4 FE 24 E0 FE A3
02F0: 83 FE B3 83 A9 69 69-03 20 B8 18 A0 18 A0 F9 17
0300: 47 77 49 39 83 37 B9-08 B8 00 12 10 77 18 E9 0A
0310: F8 83 32 3D B2 17 83-F8 D5 47 77 A9 09 53 87 49
0320: D3 80 39 FB A9 FC AA-54 10 F6 42 54 09 F6 42 54
0330: 14 F6 42 C5 F8 D5 AA-27 A9 54 09 F6 42 C5 83 76
0340: 42 83 C5 89 78 23 9F-04 BD 76 4C 83 FF 32 3D D2
0350: 3D B2 60 23 20 14 B9-D5 27 AB AC 23 80 39 64 3D
0360: D5 FB 4C C6 3D 54 10-F6 42 27 A9 AA 89 78 C5 35
0370: 23 60 DF 04 BF 15 35-23 50 39 C6 8B 27 96 8B 97
0380: F6 8B A7 E6 8B A5 76-8B B5 76 8D 64 8B B8 3F F0
0390: 37 A0 D0 96 8B F0 37-A0 E8 8F BD 50 27 AA AB AE
03A0: A9 FD 03 08 AD 39 54-E0 DA AA BC 08 FB 6B AB FA
03B0: 7A AA B6 BB D3 80 AA-FB D3 05 AB EC AC 1E FE 96
03C0: A6 19 F9 37 52 A1 FA-4B 96 8B 89 78 04 C6 0A 0D
03D0: 0E 01 00 00 00 00 00-00 00 00 00 00 00 00 02
03E0: 03 04 05 06 07 08 09-00 00 00 00 00 00 00 00
03F0: 00 00 00 00 00 00 0F-10 11 12 13 14 0B 0C DC 1D
```

Tab. 1. Obsah paměti EPROM 8748

```
0000: C0 F9 A4 B0 99 92 82-F8 80 98 88 83 C6 A1 86 FF
0010: 40 79 24 30 19 12 02-78 00 18 08 03 46 21 06 7F
```

Tab. 2. Obsah paměti PROM 74188

LOC	OBJ	LINE	SOURCE STATEMENT
		1	\$pagewidth(96) pagelength(70)
		2	\$title (Dalkove ovladani pro TVP COLOR 419, 8.1.1990)
		3	
		4	*****
		5	;
		6	;* D E K L A R A C E K O N S T A N T A P R O M E N N Y C H *
		7	;
		8	*****
		9	
= 0001		10	B0 equ 00000001B ;definice bitu 0 az 7
= 0002		11	B1 equ 00000010B
= 0004		12	B2 equ 00000100B
= 0008		13	B3 equ 00001000B
= 0010		14	B4 equ 00010000B
= 0020		15	B5 equ 00100000B
= 0040		16	B6 equ 01000000B
= 0080		17	B7 equ 10000000B
		18	
= 0008		19	SynPulz equ B3 ;vstup synchronizacnich pulzu
= 0010		20	DACH1 equ B4 ;ovladani hlasitosti (H),
= 0020		21	DACJas equ B5 ; jas (J),
= 0040		22	DACSyt equ B6 ; sytosti (S)
= 0080		23	Vypinac equ B7 ;a rele pro pripojeni site
		24	
= 0003		25	Cas equ 3 ;casova jednotka pro vnitri citac
= 0004		26	tD equ 1024*2/32/5/Cas ;casova jednotka prijmu znaku (1 ms)
= 0015		27	tD0 equ 5*tD + 1 ;delka bitu 0
= 001D		28	tD1 equ 7*tD + 1 ;delka bitu 1
= 0019		29	tD0tD1 equ (tD0 + tD1)/2 ;rozhodovaci uroven pro 0/1
= 002B		30	tS equ tD1*3/2 ;odstup mezi instrukcni slovy
= 0004		31	ms200 equ 2*256/157 + 1 ;hodnota pro mereni cca 200 ms
= 0052		32	PrepInt equ 50*256/157 + 1 ;interval pro automaticke prepini
		33	;kanalu (5 s)
= 0040		34	maxHJS equ 64 ;maximalni hodnota H, J, S
= 0013		35	DfIH equ maxHJS*30/100 ;implicitni hodnoty H,
= 0020		36	DfIJ equ maxHJS*50/100 ; J
= 0020		37	DfIS equ maxHJS*50/100 ; a S (30% a 50% maximalni hodnoty)
		38	
= 0078		39	Tma equ B6 + B5 + B4 + B3
= 0050		40	ChrA equ 0Ah * 8 ;znak 'A'
		41	
= 0005		42	LastCh equ 5 ;u nas se da chytit jen 5 programu
= 0005		43	dKlice equ 5 ;delka identifikacniho klice
= 03FB		44	KlicROM equ 400H - dKlice ;a jeho umisteni
		45	
		46	;
		47	\$DsegInt
		48	;deklarace vnitri RAM
		49	;
D 0000 0001		50	R00: ds 1 ;pracovni registr
D 0001 0001		51	R01: ds 1 ;citac bitu v prijmane instrukci
D 0002 0001		52	R02: ds 1 ;citac pro mereni odstupu mezi bity
D 0003 0001		53	R03: ds 1 ;citac do 61.44 ms (256*240 us)
D 0004 0001		54	R04: ds 1 ;citac do 15.73 s (256*61.44 ms)
D 0005 0001		55	R05: ds 1 ;citac mereni odstupu mezi instrukcemi
D 0006 0001		56	R06: ds 1 ;prijmana instrukce
D 0007 0001		57	R07: ds 1 ;stav
		58	
0018		59	org 18H
		60	
D 0018 0001		61	aktHJS: ds 1 ;ukazatel na HJS ; pracovni
D 0019 0001		62	R11: ds 1 ;pracovni ukazatel HJS ; citac casu #1
D 001A 0001		63	RgHlas: ds 1 ;H (hlasitost) ; citac casu #2
D 001B 0001		64	RgJas: ds 1 ;J (jas) ; citac casu #3

LOC	OBJ	LINE	SOURCE STATEMENT
D 001C	0001	65	RgSyt: ds 1 ;S (sytoš) ; citac casu #4
D 001D	0001	66	StTlac: ds 1 ;stav tlacitek
D 001E	0001	67	R16: ds 1 ;citac do 64 (pro DACy)
D 001F	0001	68	R17: ds 1 ;oblast pro uklid akumulatoru
		69	
D 0020	0003	70	HJS1: ds 3 ;HJS pro kanal 1
D 0023	0003	71	HJS2: ds 3 ;HJS pro kanal 2
D 0026	0003	72	HJS3: ds 3 ;HJS pro kanal 3
D 0029	0003	73	HJS4: ds 3 ;HJS pro kanal 4
D 002C	0003	74	HJS5: ds 3 ;HJS pro kanal 5
D 002F	0003	75	HJS6: ds 3 ;HJS pro kanal 6
D 0032	0003	76	HJS7: ds 3 ;HJS pro kanal 7
D 0035	0003	77	HJS8: ds 3 ;HJS pro kanal 8
		78	
D 0038	0001	79	Kanala: ds 1 ;aktualni (zapnuty) kanal
D 0039	0001	80	Kanal?: ds 1 ;kanal pri stisku "?"
D 003A	0005	81	KlicRAM: ds dKlice ;klic pro test platnosti dat v RAM
D 003F	0001	82	LastIns: ds 1 ;minula instrukce
		83	
		84	; F0 = 0 ... normalni rezim F1 = 0 ... normalni rezim
		85	; F0 = 1 ... utlum hlasitosti F1 = 1 ... pohotovostni stav
		86	
		87	
		88	
		89	-----
		90	Stavove slovo:
		91	-----
		92	0 1
		93	B0 vypnuto vypinacem vypnuto DO
		94	B1 opakovana instrukce
		95	B2 stisk "?" - mer cas
		96	B3 skoncil vysilani
		97	B4 zmena citace S
		98	B5 programuje se cas
		99	B6 cas je naprogramovan
		100	-----
		101	odpovidajici bity stavoveho slova :
		102	-----
= 0001		103	VypDO equ B0
= 0002		104	OpakIns equ B1
= 0004		105	Stisk? equ B2
= 0008		106	ZrniTo equ B3
= 0010		107	ZmenaT equ B4
= 0020		108	PrgCas equ B5
= 0040		109	CasNap equ B6
		110	
		111	\$cseg
		112	*****
		113	;
		114	* Z A C A T E K H L A V N I H O P R O G R A M U *
		115	;
		116	*****
		117	
0000 C5		118	sel rb0 ;hned na uvod proved
0001 6475		119	jmp TESTuC ;testy uC, RAM a ROM
		120	
		121	-----
0003		122	org 3
		123	
0003 147F		124	ExtInt: ;vnejsi preruseni by se nemelo vubec
0005 0400		125	call ired ;vyskytnout, pokud se to sem
		126	jmp 0 ;dostalo, tak zacni zase od zacatku
		127	
		128	\$eject

LOC	OBJ	LINE	SOURCE STATEMENT
		129	*****
		130	;* *
		131	;* O B S L U H A V N I T R N I H O C I T A C E *
		132	;* *
		133	*****
		134	
0007		135	org 7
		136	
		137	CntInt: ;preruseni chodi po 240 mikrosekundach
0007 D5		138	sel rbl ; (= 96*2.5 us, delka obsluzneho
0008 AF		139	mov r7,a ; programu nesmi prekrocit 96 byte)
0009 160B		140	jtf \$ + 2
000B 23FD		141	mov a,#-Cas
000D 62		142	mov t,a ; citac se nastavuje na -3 (3 * 32 = 96)
000E 766F		143	jfi Citace ; v pohotovostnim stavu jen hlidej cas
0010 C5		144	sel rbl
0011 FF		145	mov a,r7 ; je-li ve stavovem slove nahozen
0012 D5		146	sel rbl ; bit "ZrniTo", nebudou se obsluhovat
0013 725B		147	jb3 Zrneni ; D/A prevodniky, ale hlida se cas
		148	;
		149	DACy: ;jinak proved obsluhu D/A prevodniku:
0015 1E		150	inc r6 ; zvys citac do 64 pro kmitani do DACu
0016 FE		151	mov a,r6
0017 533F		152	anl a,#maxHJS - 1 ; je-li <> 0, pockej, vsechny
0019 9635		153	jnz Pockej ; cesty musi trvat presne stejne dlouho
		154	
		155	;
		156	NewHJS: ;je-li citac nulovy, obnov hodnoty HJS
001B F1		157	mov a,@r1 ; pro dalsi cyklus v R2, R3 a R4
001C AA		158	mov r2,a
001D 19		159	inc r1
001E F1		160	mov a,@r1
001F AB		161	mov r3,a
0020 19		162	inc r1
0021 F1		163	mov a,@r1
0022 AC		164	mov r4,a
		165	Kmitej: ;kmitani do DACu:
0023 B62D		166	jf0 Utlum ; v rezimu "utlum" je hlasitost na nule
		167	;
		168	Hlasitost: ;rizeni hlasitosti
0025 FA		169	mov a,r2 ;=====
0026 C630		170	jz Hlas0
		171	Hlas1:
0028 9ABF		172	anl p2,#not DACH1
002A CA		173	dec r2
002B 0441		174	jmp Jas
		175	
		176	;
		177	Utlum: ;v rezimu "utlum" je hlasitost na nule
002D 00		178	nop ;pruchod touto cestou musi ale trvat
002E 0430		179	jmp Hlas0 ; stejne dlouho jako pri "kmitani"
		180	
		181	Hlas0:
0030 8A10		182	orl p2,#DACH1
0032 00		183	nop
0033 0441		184	jmp Jas
		185	
		186	;
		187	Pockej: ;ted je chvilka cas, tak si pripravime
0035 F8		188	mov a,r0 ; do R1' aktHJS (program za navestim
0036 A9		189	mov r1,a ; NewHJS ocekava toto cislo v R1')
0037 B63D		190	jf0 TeckaOn ; v rezimu "utlum" se zobrazuje tecka
0039 997F		191	anl p1,#not B7
003B 0423		192	jmp Kmitej

LOC	OBJ	LINE	SOURCE STATEMENT
		193	TeckaOn:
003D 8980		194	orl p1,#B7
003F 0423		195	jmp Knitej
		196	
		197	-----
		198	Jas: ;rizeni jas
0041 FB		199	mov a,r3 ;=====
0042 C649		200	jz Jas0
		201	Jas1:
0044 9ADF		202	anl p2,#not DACjas
0046 CB		203	dec r3
0047 044E		204	jmp Sytost
		205	Jas0:
0049 8A20		206	orl p2,#DACjas
004B 00		207	nop
004C 044E		208	jmp Sytost
		209	
		210	-----
		211	Sytost: ;rizeni sytosti
004E FC		212	mov a,r4 ;=====
004F C656		213	jz Syt0
		214	Syt1:
0051 9ABF		215	anl p2,#not DACsynt
0053 CC		216	dec r4
0054 046F		217	jmp Citace
		218	Syt0:
0056 8A40		219	orl p2,#DACsynt
0058 00		220	nop
0059 046F		221	jmp Citace
		222	-----
		223	Zrneni:
005B FA		224	mov a,r2 ;pokud prestaly chodit synchronizacni
005C CA		225	dec r2 ;pulzy (obrazovka zrn), prestali uz
005D 966F		226	jnz Citace ;asi vysilat. Hlasitost, jas
005F FB		227	mov a,r3 ;i sytost jsou na nule
0060 CB		228	dec r3 ;a registry R2, R3, R4 se vyuzivaji
0061 966F		229	jnz Citace ;pro mereni intervalu 5 minut.
0063 FC		230	mov a,r4 ;Pokud do teto doby divak
0064 CC		231	dec r4 ;neprepne na kanal, který jeste
0065 966F		232	jnz Citace ;vysila, dojde k vypnuti TVP.
		233	Spisladce:
0067 C5		234	sel rb0
0068 BF01		235	mov r7,#VypDO
006A 8AFF		236	orl p2,#-1
006C 8978		237	orl p1,#Tma
006E B5		238	cpl f1
		239	
		240	-----
		241	Citace: ;zvys vsechny citace:
006F C5		242	sel rb0
0070 1A		243	inc r2 ;R2 meri delku instrukce
0071 1B		244	inc r3 ;R3 a R4 meri cas
0072 FB		245	mov a,r3
0073 967B		246	jnz Navrat
0075 1C		247	inc r4 ;pri zmene tohoto citace si poznacime
0076 CD		248	dec r5 ;do stavoveho slova, ze je cas
0077 2310		249	mov a,#ZmenaT ;pro testovani tlacitek (tlacitka se
0079 4F		250	orl a,r7 ;sleduji jen obcas)
007A AF		251	mov r7,a
		252	Navrat:
007B 7680		253	jf1 Tecka
		254	Return:
007D D5		255	sel rb1 ;nejdrive se musi obnovit obsah
007E FF		256	mov a,r7 ;akumulatoru

LOC	OBJ	LINE	SOURCE STATEMENT
		257	iret:
007F 93		258	retr ;a pak je mozny navrat z preruseni
		259	
		260	;
		261	Tecka: ;v pohotovostnim stavu se:
0080 FF		262	mov a,r7
0081 B27D		263	jb5 Return ;nedela nic (pokud se programuje cas),
0083 D28A		264	jb6 TstCas ;testuji citace naprogramovaneho casu
0085 FC		265	mov a,r4 ;(je-li cas naprogramovan) nebo pomalu
0086 92B5		266	jb4 Tecka0 ;blika tecka
0088 04B1		267	jmp Tecka1
		268	
		269	TstCas: ;aktualizace a test citacu
009A D5		270	sel rb1 ;naprogramovaneho casu
009B 19		271	inc r1
009C F9		272	mov a,r1
009D D3FA		273	xrl a,#250 ;250 * 240 us = 60 ms
009F 96AD		274	jnz Blikej
0091 A9		275	mov r1,a
0092 1A		276	inc r2
0093 FA		277	mov a,r2
0094 D3FA		278	xrl a,#250 ;250 * 60 ms = 15 s
0096 96AD		279	jnz Blikej
0098 AA		280	mov r2,a
0099 CC		281	dec r4
009A FC		282	mov a,r4
009B 96AD		283	jnz Blikej
009D FB		284	mov a,r3
009E CB		285	dec r3
009F 96AD		286	jnz Blikej
00A1 C5		287	sel rb0 ;kdyz uz naprogramovany cas konecne
00A2 AF		288	mov r7,a ;vyprsel, vynuluj stavove slovo,
00A3 A5		289	clr f1 ;zrus pohotovostni stav, zapni stroje
00A4 9A7F		290	anl p2,#not Vypinac ;a kanal, který tam byl
00A6 B838		291	mov r0,#KanalA
00A8 F0		292	mov a,@r0
00A9 54F4		293	call KanalOn
00AB 047D		294	jmp Return
		295	
		296	Blikej: ;v pohotovostnim stavu,
00AD C5		297	sel rb0 ;pri naprogramovane case, blika tecka
00AE FC		298	mov a,r4 ;rychleji
00AF 52B5		299	jb2 Tecka0
		300	Tecka1:
00B1 8980		301	orl p1,#B7
00B3 047D		302	jmp Return
		303	Tecka0:
00B5 997F		304	anl p1,#not B7
00B7 047D		305	jmp Return
		306	
		307	;
		308	orlSS: ;nahozeni bitu ve stavovem slove
00B9 35		309	dis tcnti
00BA 4F		310	orl a,r7
00BB 04BF		311	jmp ZmenSS
		312	
		313	;
		314	anlSS: ;zruseni bitu ve stavovem slove
00BD 35		315	dis tcnti
00BE 5F		316	anl a,r7
		317	ZmenSS: ;zmena stavoveho slova se musi
00BF AF		318	mov r7,a ;provadet se zakazanym prerusenim
00C0 25		319	en tcnti ;(behem preruseni se ve stavovem
00C1 1607		320	jtf CntInt ;slove take cvici s bity); pak se ale

LOC	OBJ	LINE	SOURCE STATEMENT
00C3	83	321	ret ;musi pohlídat, aby preruseni neuteklo
		322	
		323	;
		324	JmpP0: ;cteni jednoho byte z bezne stranky
00C4	A3	325	movp a,@a ;procedura musi lezet v 0. strance
00C5	83	326	ret
		327	
		328	*****
		329	*****
		330	***** POKRACOVANI HLAVNIHO PROGRAMU *****
		331	*****
		332	*****
		333	
		334	Start:
00C6	B83A	335	mov r0,#KlicRAM
00C8	B9FB	336	mov r1,#KlicROM
00CA	BA05	337	mov r2,#dKlice
00CC	85	338	clr f0
00CD	95	339	cpl f0
		340	TestKlice: ;napred testuj, zda jsou data v RAM
00CE	F9	341	mov a,r1 ;platna (soucasne se do RAM ulozi
00CF	E3	342	movp3 a,@a ;hledany identifikacni klic)
00D0	20	343	xch a,@r0
00D1	D0	344	xrl a,@r0
00D2	C6D5	345	jz NextChar
00D4	85	346	clr f0 ;pri nesouhlasu shod flag F0
		347	NextChar:
00D5	18	348	inc r0
00D6	19	349	inc r1
00D7	EACE	350	djnz r2,TestKlice
		351	
00D9	B81D	352	mov r0,#StTlac ;pak inicializuj stav tlacitek
00DB	08	353	ins a,Bus
00DC	A0	354	mov @r0,a
00DD	B6EE	355	jf0 RAMok ;pri spravne RAM preskoc
00DF	7405	356	call Tlacitka ;jeji inicializaci
00E1	B838	357	mov r0,#Kanala ;a inicializaci cisla kanalu
00E3	A0	358	mov @r0,a
		359	
		360	;
		361	InitRAM: ;do prazdne RAM uloz implicitni
00E4	B820	362	mov r0,#HJS1 ;hodnoty H (hlasitosti), J (jasu)
00E6	B908	363	mov r1,#8 ;a S (sytylosti) pro vseh 8 kanalu
		364	InitHJS:
00E8	541C	365	call Default
00EA	E9E8	366	djnz r1,InitHJS
00EC	27	367	clr a
00ED	AF	368	mov r7,a
		369	
		370	;
		371	RAMok: ;je-li RAM "spravna",
00EE	B820	372	mov r0,#HJS1
00F0	B918	373	mov r1,#8 * 3
		374	Kontrola: ;kontroluje se, zda udaje v RAM
00F2	F0	375	mov a,@r0 ;nepresahuji maximalni hodnotu
00F3	03BF	376	add a,#-(maxHJS + 1)
00F5	E6F9	377	jnc DalsiB
00F7	B020	378	mov @r0,#dflJ
		379	DalsiB:
00F9	18	380	inc r0
00FA	E9F2	381	djnz r1,Kontrola
		382	
00FC	B838	383	mov r0,#Kanala ;zkontroluj i cislo kanalu
00FE	F0	384	mov a,@r0

LOC	OBJ	LINE	SOURCE STATEMENT
00FF 5307		385	anl a,#7
0101 A0		386	mov @r0,a
		387	
0102 8AFF		388	orl p2,#-1 ;nechej stroj vypnuty,
0104 A5		389	clr f1 ;ale F1 nastav na "normalni rezim"
0105 95		390	cpl f0 ;pro dalsi test potrebujeme F0 otocit
0106 B611		391	jf0 Pohotovost ;pri "vadne" RAM
0108 FF		392	mov a,r7 ;nebo pri vypnuti pomoci DO zustan
0109 1211		393	jb0 Pohotovost ;v pohotovostnim stavu, jinak (RAM
		394	Beh: ;je O.K. a stroj byl vypnut vypinacem)
010B F0		395	mov a,@r0 ;zapni napajeni a naposled
010C 54F4		396	call KanalOn ;sledovany kanal
010E 9A7F		397	anl p2,#not Vypinac
0110 B5		398	cpl f1
		399	Pohotovost:
0111 B5		400	cpl f1 ;nastav spravny rezim cinnosti
0112 85		401	clr f0 ;(pohotovost nebo beh), F0 od tohoto
0113 FF		402	mov a,r7 ;okamziku bude ridit utlum
0114 5301		403	anl a,#VypDO
0116 AF		404	mov r7,a ;pak inicializuj stav,
0117 25		405	en tcnti ;a spust mereni casu
0118 55		406	strt t
		407	*****
		408	
		409	Prijem: ;PRIJEM INSTRUKCE
0119 5426		410	call CekejInstrukci
011B 97		411	clr c ;po precteni instrukce
011C 67		412	rrc a ;presun start-bit do CY a vyhodnot jej
011D 2620		413	jnt0 StBit0 ;dle prepinae na vstupu T0
		414	StBit1: ;start-bit na byt 1 (jako CY)
011F A7		415	cpl c ;nyni jako NC
		416	StBit0: ;start-bit na byt 0 (jako NC)
0120 F619		417	jc Prijem ;je-li tedy CY = 1, neni to instrukce
0122 B93F		418	mov r1,#LastIns ;pro nas - ostatni se zpracovavaji,
0124 21		419	xch a,@r1 ;ale pro jistotu jen kdyz prijdou
0125 D1		420	xrl a,@r1 ;dve stejne instrukce za sebou
0126 9619		421	jnz Prijem
0128 F1		422	mov a,@r1 ;zpracovavame jen instrukce 0 az 47,
0129 03D0		423	add a,#-DelTab ;instrukce 48 a dal nas uz nezajimaji
012B F1		424	mov a,@r1
012C F619		425	jc Prijem
012E 3438		426	call Zpracovani
0130 2302		427	mov a,#OpakIns
0132 14B9		428	call orlSS
0134 BD04		429	mov r5,#ms200
0136 2419		430	jmp Prijem
		431	*****
		432	
		433	Zpracovani: ;ZPRACOVANI INSTRUKCE
0138 03CE		434	add a,#DekodTbl
013A B3		435	movp3 a,@a ;prekoduji instrukci
013B A8		436	mov r0,a ;a uschovej si ji, její kod se nam
013C 7640		437	jf1 TstZlom ;jeste bude hodit; v pohotovostnim
013E 2445		438	jmp Proved ;rezimu se provadeji jen nektere
		439	TstZlom: ;instrukce
0140 03F2		440	add a,#-Zlom
0142 F6DF		441	jc ZadnaInstrukce
0144 F8		442	mov a,r0
		443	Proved:
0145 C8		444	dec r0
0146 0349		445	add a,#InsTbl
0148 B3		446	jmp @a ;proved instrukci
		447	
		448	*****

LOC	OBJ	LINE	SOURCE STATEMENT
		449	
		450	InstTbl: ;skokova tabulka po prekodovani
		451	;instrukci na nas novy kod (#n)
0149 DF		452	db ZadnaInstrukce ;neznama instrukce (#0)
014A 6B		453	db ?Prog ;projeti vseh kanalu (#1)
014B F5		454	db ZapPrg ;zapnuti pristroje a kanalu #1 (#2)
014C F5		455	db ZapPrg ;zapnuti pristroje a kanalu #2 (#3)
014D F5		456	db ZapPrg ;zapnuti pristroje a kanalu #3 (#4)
014E F5		457	db ZapPrg ;zapnuti pristroje a kanalu #4 (#5)
014F F5		458	db ZapPrg ;zapnuti pristroje a kanalu #5 (#6)
0150 F5		459	db ZapPrg ;zapnuti pristroje a kanalu #6 (#7)
0151 F5		460	db ZapPrg ;zapnuti pristroje a kanalu #7 (#8)
0152 F5		461	db ZapPrg ;zapnuti pristroje a kanalu #8 (#9)
0153 8F		462	db Normovani ;normovani H, J, S (#10)
0154 B2		463	db AFCplus ;zahajeni/ukonceni, pripadne (#11)
0155 E4		464	db AFCminus ;zruseni programovani casu (#12)
0156 B8		465	db UtlumZmena ;zmena rezimu "utlum", zapnuti (#13)
0157 5E		466	db Vypnuti ;vypnuti stroje (#14)
0158 9A		467	db Hplus ;zvyseni hlasitosti (#15)
0159 A1		468	db Hminus ;snizeni hlasitosti (#16)
015A B3		469	db Jplus ;zvyseni jasu (#17)
015B B9		470	db Jminus ;snizeni jasu (#18)
015C BF		471	db Splus ;zvyseni sytosti (#19)
015D D1		472	db Sminus ;snizeni sytosti (#20)
		473	
= 000E		474	Zlom equ 14 ;instrukce, ktera uz se neprovadi
		475	;v rezimu "pohotovost"
		476	
		477	;*****
		478	Vypnuti: ;VYPNUTI PRISTROJE POMOCI DO
015E A5		479	clr f1
015F B5		480	cpl f1 ;nastav pohotovostni stav
0160 85		481	clr f0 ;zrus utlum, aby mel po zapnuti
0161 35		482	dis tcnti ;definovany stav
0162 8AFF		483	orl p2, #-1 ;vypni stroje, zhasni display
0164 8978		484	orl p1, #Tma
0166 2301		485	mov a, #VypDO ;vypnuti se zapise do stavoveho slova
0168 14BF		486	call ZmenSS ;ostatni bity stavoveho slova vynuluj
016A 83		487	ret
		488	
		489	;-----
		490	?Prog: ;CO JE DNES NA PROGRAMU ? / CIFRA 0
016B FF		491	mov a, r7 ;postupne projde vsechny obsazene
016C 76B6		492	jf1 NaprogramujCas ;kanaly a vrati se na puvodni kanal,
016E 32DF		493	jbl ZadnaInstrukce ;ale jen pri prvni stisku tlacitka
0170 5286		494	jbl ?Stop ;je-li uz rezim "?" nastaven, zrus jej
0172 2304		495	mov a, #Stisk?
0174 14B9		496	call orlSS ;jinak se tento rezim zahaji
0176 27		497	clr a
0177 AC		498	mov r4, a ;inicializace citace pro mereni casu
0178 B838		499	mov r0, #KanalA ;mezi prepinanim kanalu
017A F0		500	mov a, @r0
017B 03FC		501	add a, # -(LastCh - 1)
017D 27		502	clr a ;je-li zapnut "vysoky" nebo "posledni"
017E F682		503	jc NxtChnl ;kanal, prepni na kanal #1
0180 F0		504	mov a, @r0 ;jinak zapni dalsi kanal v poradí
0181 17		505	inc a
		506	NxtChnl:
0182 18		507	inc r0 ;novy kanal se ulozi na Kanal?
0183 A0		508	mov @r0, a
0184 44F4		509	JMP KanalOn
		510	; ret ;pro usporu mista a casu
		511	?Stop:
0186 B839		512	mov r0, #Kanal?

LOC	OBJ	LINE	SOURCE STATEMENT
0188	F0	513	mov a,@r0 ;rezim "?" se zrusi zachovanim prave
0189	C8	514	dec r0 ;sledovaneho kanalu
018A	A0	515	mov @r0,a
018B	23FB	516	mov a,#not Stisk? ;a vynulovanim prislusneho bitu
018D	04BD	517	JMP anlSS ;ve stavovem slove,
		518	; ret
		519	
		520	-----
		521	Normovani: ;NORMOVANI H, J, S / CIFRA 9
018F	FF	522	mov a,r7 ;kdyz to nezrni,
0190	70B6	523	jfi NaprogramujCas
0192	72DF	524	jbs ZadnaInstrukce
0194	B818	525	mov r0,#aktHJS ;nastavi se implicitni hodnoty
0196	F0	526	mov a,@r0 ;pro dany kanal
0197	A8	527	mov r0,a
0198	441E	528	JMP DfItJS
		529	; ret
		530	
		531	-----
		532	Hplus: ;ZVYSENI HLASITOSTI
019A	B818	533	mov r0,#aktHJS
019C	F0	534	mov a,@r0
019D	B6A8	535	jfo ZrusUtlum ;zmena hlasitosti v rezimu "utlum"
019F	24C4	536	jmp ZvysHJS ;zrusi tento rezim
		537	
		538	-----
		539	Hminus: ;SNIZENI HLASITOSTI
01A1	B818	540	mov r0,#aktHJS
01A3	F0	541	mov a,@r0
01A4	B6A8	542	jfo ZrusUtlum ;zmena hlasitosti v rezimu "utlum"
01A6	24D6	543	jmp SnizHJS ;zrusi tento rezim
		544	
		545	-----
		546	ZrusUtlum:
01A8	A8	547	mov r0,a ;pokud pri utlumu manime hlasitost
01A9	FF	548	mov a,r7 ;a nezrni to,
01AA	72DF	549	jbs ZadnaInstrukce
01AC	27	550	clr a
01AD	A0	551	mov @r0,a ;nastavi se hlasitost na nulu
01AE	B81A	552	mov r0,#RgHlas
01B0	A0	553	mov @r0,a
01B1	85	554	clr f0 ;a rezim utlum se zrusi
01B2	83	555	ret
		556	
		557	-----
		558	Jplus: ;ZVYSENI JASU
01B3	B818	559	mov r0,#aktHJS
01B5	F0	560	mov a,@r0
01B6	17	561	inc a
01B7	24C4	562	jmp ZvysHJS
		563	
		564	-----
		565	Jminus: ;SNIZENI JASU
01B9	B818	566	mov r0,#aktHJS
01BB	F0	567	mov a,@r0
01BC	17	568	inc a
01BD	24D6	569	jmp SnizHJS
		570	
		571	-----
		572	Spplus: ;ZVYSENI SYTOSTI
01BF	B818	573	mov r0,#aktHJS
01C1	F0	574	mov a,@r0
01C2	17	575	inc a
01C3	17	576	inc a

LOC	OBJ	LINE	SOURCE STATEMENT
		577	ZvysHJS:
01C4 A8		578	mov r0,a ;H,J,S se meni, jen kdyz dany
01C5 FF		579	mov a,r7 ;kanal vysila
01C6 72DF		580	jb3 ZadnaInstrukce
01C8 F0		581	mov a,@r0
01C9 17		582	inc a
01CA D2CE		583	jb6 HJSmax
01CC A0		584	mov @r0,a
01CD 83		585	ret
		586	HJSmax:
01CE B040		587	mov @r0,#maxHJS
01D0 83		588	ret
		589	
		590	;
		591	Sminus: ;SNIZENI SYTOSTI
01D1 B818		592	mov r0,#aktHJS
01D3 F0		593	mov a,@r0
01D4 17		594	inc a
01D5 17		595	inc a
		596	SnizHJS:
01D6 A8		597	mov r0,a ;H,J,S se meni, jen kdyz dany
01D7 FF		598	mov a,r7 ;kanal vysila
01D8 72DF		599	jb3 ZadnaInstrukce
01DA F0		600	mov a,@r0
01DB C6DF		601	jz ZadnaInstrukce
01DD 07		602	dec a
01DE A0		603	mov @r0,a
		604	ZadnaInstrukce:
01DF 83		605	ret
		606	
		607	;
		608	JmpP1:
01E0 A3		609	movp a,@a ;cteni jednoho byte z bezne stranky
01B1 83		610	ret ;procedura musi lezet v 1. strance
		611	;
		612	
		613	AFCplus: ;ZACATEK/KONEC PROGRAMOVANI CASU
01E2 6449		614	jmp AFCp
		615	
		616	;
		617	AFCminus: ;ZRUSENI NAPIROGRAMOVANEHO CASU
01E4 643F		618	jmp AFCm
		619	
		620	;
		621	NaprogramujCas:
01E6 6412		622	jmp ProgramovaniCasu
		623	
		624	;
		625	UtlumZmena: ;ZMENA REZIMU "UTLUM"
01E8 FF		626	mov a,r7
01E9 32DF		627	jb1 ZadnaInstrukce ;utlum se zmeni jen pri prvnim stisku
01EB 72DF		628	jb3 ZadnaInstrukce ;tlacitka a kdyz to
01ED 95		629	cpl f0 ;nezrni
01EE 52DF		630	jb2 ZadnaInstrukce ;mimo rezim "?" se
01F0 B838		631	mov r0,#Kanala
01F2 F0		632	mov a,@r0 ;stiskem tohoto tlacitka soucasne
01F3 24FF		633	jmp ZapniTo ;zapina stroj
		634	
		635	;
		636	ZapPrg: ;ZAPNUTI PRISTROJE / CIPRY 1 - 8
01F5 FF		637	mov a,r7 ;podivej se napred do stavoveho slova:
01F6 32DF		638	jb1 ZadnaInstrukce
01F8 B2B6		639	NaprogramujCas ;bud pujdeme naprogramovat cas noveho
01FA C8		640	dec r0 ;zapnuti nebo zapneme stroje a vhodny

LOC	OBJ	LINE	SOURCE STATEMENT
01FB 2338	641	mov	a, #KanalA ; kanal (jeho cislo je v R0 , ale
01FD 28	642	xch	a, r0 ; o 1 vetsi)
01FE A0	643	mov	@r0, a
	644	ZapniTo:	
01FF 54F4	645	call	KanalOn
0201 A5	646	clr	f1
0202 9A7F	647	anl	p2, #not Vypinac
0204 239A	648	mov	a, #not (PrgCas or CasNaprg or Stisk? or VypDO)
0206 14BD	649	call	anlSS ; pak jeste musime trochu uklidit
0208 83	650	ret	; ve stavovem slove
	651		
	652	;*****	
	653	; K o n e c z p r a c o v a n i (toto uz smi byt v jine strance)*	
	654	;*****	
	655		
	656	Add12:	; R3', R4' := R3', R4' + R1', R2'
0209 2C	657	xch	a, r4
020A 6A	658	add	a, r2
020B 2C	659	xch	a, r4
020C 2B	660	xch	a, r3
020D 79	661	addc	a, r1
020E 2B	662	xch	a, r3
020F 83	663	ret	
	664		
	665	;*****	
	666	Mult4:	; nasobeni R3', R4' ctyrmi
0210 5414	667	call	Mult2
0212 F61B	668	jc	Ovrflw ; pri pretecani hned skonci
	669	Mult2:	; nasobeni R3', R4' dvemi
0214 97	670	clr	c
0215 2C	671	xch	a, r4
0216 F7	672	rlc	a
0217 2C	673	xch	a, r4
0218 2B	674	xch	a, r3
0219 F7	675	rlc	a
021A 2B	676	xch	a, r3
	677	Ovrflw:	; sem skaceme pri pretecani
021B 83	678	ret	
	679	;-----	
	680		
	681	Default:	; nastaveni implicitnich hodnot H, J, S
021C B013	682	mov	@r0, #DflH ; pro jeden kanal
	683	DfltJS:	
021E 18	684	inc	r0
021F B020	685	mov	@r0, #DflJ
0221 18	686	inc	r0
0222 B020	687	mov	@r0, #DflS
0224 18	688	inc	r0
0225 83	689	ret	
	690		
	691	;-----	
	692		
	693	CekajInstrukci:	; CTRNI INSTRUKCE
0226 86A2	694	jnl	StartBit ; cekaj na start-bit a obcas testuj
0228 FD	695	mov	a, r5 ; casovy odstup mezi instrukcemi,
0229 962F	696	jnz	MeziCas
022B 23FD	697	mov	a, #not OpakIns
022D 14BD	698	call	anlSS
	699	MeziCas:	
022F FF	700	mov	a, r7 ; sleduj i zmenu tlacitek
0230 9263	701	jb4	ZprTlac ; a bylo-li stisknuto tlacitko "?",
0232 5236	702	jb2	ZmenaK ; testuj i cas pro prepnuti kanalu
0234 4426	703	jmp	CekajInstrukci ; jinak zustan v cekaci smyce
	704	ZmenaK:	; testuj, zda je mozno prepnout

LOC	OBJ	LINE	SOURCE STATEMENT
0236	FC	705	mov a,r4 ;na dalsi kanal
0237	03AB	706	add a,#-PrepInt
0239	F644	707	jc DalsiPanNaHoleni
023B	FF	708	mov a,r7 ;pri zrneni se ceka kratši dobu
023C	37	709	cpl a
023D	7226	710	jb3 CekejInstrukci
023F	FC	711	mov a,r4
0240	03E5	712	add a,#-PrepInt/3
0242	E626	713	jnc CekejInstrukci
		714	DalsiPanNaHoleni:
0244	AC	715	mov r4,a ;uplynul-li nektery z obou
0245	B839	716	mov r0,#Kanal? ;sledovanych casu, nastav
0247	10	717	inc @r0 ;zase registr pro mereni casu
0248	F0	718	mov a,@r0 ;a priprav cislo dalsiho kanalu
0249	03FB	719	add a,#-LastCh
024B	F0	720	mov a,@r0 ;je-li to nyní "nizky" kanal,
024C	E658	721	jnc NxtCh1 ; prepni jej
024E	C8	722	dec r0 ;jinak se podivej, odkud jsme vysli
024F	F0	723	mov a,@r0
0250	03FB	724	add a,#-LastCh
0252	27	725	clr a ;a kdyz to bylo z "nizkeho" kanalu,
0253	E656	726	jnc NxtCh2 ;zapni kanal #0, jinak prepni
0255	F0	727	mov a,@r0 ;na vychozi kanal - tady to skoncuje
		728	NxtCh2:
0256	18	729	inc r0
0257	A0	730	mov @r0,a
		731	NxtCh1:
0258	54F4	732	call KanalOn
025A	B839	733	mov r0,#Kanal?
025C	F0	734	mov a,@r0
025D	C8	735	dec r0
025E	D0	736	xrl a,@r0
025F	9626	737	jnz CekejInstrukci ;a pokud uz jsme zase
0261	447B	738	jmp ?Zrus ;na puvodnim kanalu, zrus rezim "?"
		739	
		740	;
		741	ZprTlac:
0263	23EF	742	mov a,#not ZmenaT ;cteni stavu tlacitek, pripadne
0265	14BD	743	call anlSS ;prepnuti kanalu pri jejich zmene
0267	B81D	744	mov r0,#StTlac ;a test existence
0269	08	745	ins a,Bus ;synchronizacnich pulzu
026A	20	746	xch a,@r0
026B	D0	747	xrl a,@r0
026C	C681	748	jz Synchr?
026E	F0	749	mov a,@r0
026F	7405	750	call Tlacitka
0271	7226	751	jb3 CekejInstrukci ;neni-li nic stisknuto, nic se nedeje,
0273	B838	752	mov r0,#KanalA ;jinak se zapamatuje novy kanal
0275	A0	753	mov @r0,a ;a provede se zapnuti stroju
0276	54F4	754	call KanalOn ;a prepnuti kanalu
0278	9A7F	755	anl p2,#not Vypinac
027A	A5	756	clr f1
		757	?Zrus:
027B	239A	758	mov a,#not (Stisk? or CasMapr or PrgCas or VypDO)
027D	14BD	759	call anlSS ;zrus bit "Stisk?" ve stavovem slove
027F	4426	760	jmp CekejInstrukci
		761	
		762	Synchr?:
0281	7626	763	jf1 CekejInstrukci
0283	0A	764	in a,p2 ;(ne)pritomnost synchronizacnich pulzu
0284	728C	765	jb3 Nesynch ;se poznamenava do stavoveho slova
0286	23F7	766	mov a,#not ZrniTo
0288	14BD	767	call anlSS
028A	4426	768	jmp CekejInstrukci

LOC	OBJ	LINE	SOURCE STATEMENT
		769	Nesynch: ;pri pokracujicim zrneni
028C 8980		770	orl p1,#B7 ;se uz nesmi delat nic,
028E FF		771	mov a,r7 ;pokud vsak zrneni prave zacalo,
028F 7226		772	jb3 CekejInstrukci ;musime stahnout hlasitost,
0291 65		773	stop tcnt ;sytoost a jas (nebudeme totiz mit
0292 4308		774	orl a,#ZrniTo ;cas ani prostor na jejich obsluhu)
0294 AF		775	mov r7,a ;a nastavit citac pro mereni 5 minut
0295 8A70		776	orl p2,#DAChl or DACjas or DACsy
0297 D5		777	sel rbl
0298 BA00		778	mov r2,#0 ;4883 = 5 * 60 s / (256 * 240 us)
029A BB13		779	mov r3,#4883 mod 256
029C BC13		780	mov r4,#4883 / 256
029E C5		781	sel rb0
029F 55		782	strt t
02A0 4426		783	jmp CekejInstrukci
		784	
		785	*****
		786	StartBit: ;zacatek zpracovani instrukce:
02A2 46A7		787	jnt1 \$ + 5 ;zpracovavame bity jen se spravnym
02A4 90		788	movx @r0,a ;poctem impulsu, jine (rusive) signaly
02A5 4426		789	jmp CekejInstrukci ;se zahazuji
02A7 90		790	movx @r0,a ;shod prerusovaci signal, bit jsme
02A8 27		791	clr a ;prijali; pak nulu
02A9 AE		792	mov r6,a ;instrukci
02AA AA		793	mov r2,a ;a interval pro mereni její delky
02AB B901		794	mov r1,#B0 ;prijde nuly (Start) bit
		795	Wbit:
02AD 86B6		796	jni JdeBit ;pri cekani sleduj, zda neni interval
02AF FA		797	mov a,r2 ;mezi bity prilis dlouhy - v tom
02B0 03D5		798	add a,#-tS ;pripade cekani ukonci a zacni
02B2 E6AD		799	jnc Wbit ;zpracovavat novou instrukci
02B4 4426		800	jmp CekejInstrukci
		801	JdeBit:
02B6 46BB		802	jnt1 \$ + 5 ;stale se zpracovavaji jen bity se
02B8 90		803	movx @r0,a ;spravnym poctem impulsu
02B9 4426		804	jmp CekejInstrukci
02BE 90		805	movx @r0,a
02BC FA		806	mov a,r2 ;je-li interval mezi bity prilis
02BD 03F6		807	add a,#-tD0/2 ;kratky, je to asi rusivy signal,
02BF E626		808	jnc CekejInstrukci ;cekej na jinou instrukci
02C1 27		809	clr a ;jinak hned zase nulu registr
02C2 2A		810	xch a,r2 ;pro mereni odstupu mezi bity
02C3 03E7		811	add a,#-tD0tD1 ;a rozhodni, zda se jedna o 0 nebo 1
02C5 E6CA		812	jnc NxBit?
02C7 F9		813	mov a,r1
02C8 4E		814	orl a,r6 ;jednickovy bit se pricte do prijimane
02C9 AE		815	mov r6,a ;instrukce, nula tam uz je
		816	NxBit?:
02CA F9		817	mov a,r1
02CB 69		818	add a,r1 ;posun na dalsi bit
02CC A9		819	mov r1,a
02CD F2D7		820	jb7 CeleSlovo
02CF 44AD		821	jmp Wbit
		822	MocBitu:
02D1 56D6		823	jti \$ + 5
02D3 90		824	movx @r0,a ;prislo moc bitu - zahod
02D4 4426		825	jmp CekejInstrukci ;tuto "instrukci"
02D6 90		826	movx @r0,a
		827	CeleSlovo:
02D7 86D1		828	jni MocBitu ;uz mam cele slovo; pockej chvili,
02D9 FA		829	mov a,r2 ;zda neprijde jeste nejaky dalsi bit
02DA 03D5		830	add a,#-tS
02DC E6D7		831	jnc CeleSlovo
02DE FE		832	mov a,r6 ;prijata instrukce ma spravny pocet

LOC	OBJ	LINE	SOURCE STATEMENT
02DF	83	833	ret ;bitu; dej ji do akumulatoru a vrat se
		834	
		835	;
		836	GetByte: ;podprogram pro cteni jednoho byte
02E0	F9	837	mov a,r1 ;z prislusne stranky EPROM:
02E1	03E4	838	add a,#JmpTbl ;k cislu stranky pricti adresu tabulky
02E3	B3	839	jmp @a ;skoku a skoc si pro byte
		840	;
		841	
		842	JmpTbl: ;Tabulka skoku (musi lezet cela
02E4	E8	843	db GtByP0 ;v jedne strance)
02E5	EB	844	db GtByP1
02E6	EE	845	db GtByP2
02E7	F1	846	db GtByP3
		847	;
		848	; vytahni adresu a skoc pro jeden byte
		849	GtByP0: ; do prislusne stranky
02E8	FE	850	mov a,r6
02E9	04C4	851	jmp JmpP0
		852	GtByP1:
02EB	FE	853	mov a,r6
02EC	24E0	854	jmp JmpP1
		855	GtByP2:
		856	JmpP2:
02EE	FE	857	mov a,r6
02EF	A3	858	movp a,@a
02F0	83	859	ret
		860	GtByP3:
02F1	FE	861	mov a,r6
02F2	E3	862	movp3 a,@a
02F3	83	863	ret
		864	;
		865	
		866	KanalOn: ;zapnuti kanalu, zobrazeni jeho cisla,
02F4	A9	867	mov r1,a ;vypocet ukazatele na odpovidajici HJS
02F5	69	868	add a,r1
02F6	69	869	add a,r1
02F7	0320	870	add a,#HJS1 ; 3*kanal + HJS1 pro ukazatel
02F9	B318	871	mov r0,#aktHJS ;na nove HJS
02FB	A0	872	mov @r0,a
02FC	18	873	inc r0
02FD	A0	874	mov @r0,a
		875	
02FE	F9	876	mov a,r1
02FF	17	877	inc a
0300	47	878	swap a
0301	77	879	rr a ; 8*kanal pro zobrazeni cisla kanalu
0302	49	880	orl a,r1
0303	39	881	outl pl,a
0304	83	882	ret
		883	
		884	;
		885	
		886	Tlacitka: ;zjisteni kodu stisknuteho tlacitka
0305	37	887	cpl a
0306	B908	888	mov r1,#8 ;8 pokusu
0308	B800	889	mov r0,#0
		890	Hledej:
030A	1210	891	jb0 Hotovo
030C	77	892	rr a
030D	18	893	inc r0
030E	E90A	894	djnz r1,Hledej
		895	Hotovo:
0310	F8	896	mov a,r0 ;vrati hodnotu 0 - 7

LOC	OBJ	LINE	SOURCE STATEMENT
0311	83	897	ret ;nebo 8, kdyz neni stisknuto nic
		898	
		899	;*****
		900	
		901	ProgramovaniCasu:
0312	323D	902	jb1 KonProg ;pokud se jedna o opakovany Pavel,
0314	B217	903	jb5 Mozno ;tak cifru ignoruj
0316	83	904	ret ;pokud uz bylo programovani zahajeno,
		905	Mozno: ;je mozno pokracovat jinak konci
0317	F8	906	mov a,r0 ;pak cifru zpracuj: je v R0
0318	D5	907	sel rb1
0319	47	908	swap a
031A	77	909	rr a ; vynasob cifru 8x pro zobrazeni,
031B	A9	910	mov r1,a
031C	09	911	in a,p1 ;zobraz ji
031D	5387	912	anl a,#not Tma
031F	49	913	orl a,r1
0320	D380	914	xrl a,#B7 ;se zmenenou teckou
0322	39	915	outl p1,a
0323	FB	916	mov a,r3 ;pak presun R3',R4' do R1',R2'
0324	A9	917	mov r1,a
0325	FC	918	mov a,r4
0326	AA	919	mov r2,a
0327	5410	920	call Mult4 ; * 4
0329	F642	921	jc ZrusPrg ;pri zadani prilis velkeho casu zrus
032B	5409	922	call Add12 ;programovaci rezim
032D	F642	923	jc ZrusPrg ; * 5
032F	5414	924	call Mult2 ; * 10
0331	F642	925	jc ZrusPrg
0333	C5	926	sel rb0
0334	F8	927	mov a,r0
0335	D5	928	sel rb1
0336	AA	929	mov r2,a
0337	27	930	clr a
0338	A9	931	mov r1,a
0339	5409	932	call Add12 ; 10*y + x
033B	F642	933	jc ZrusPrg
		934	KonProg:
033D	C5	935	sel rb0
033E	83	936	ret
		937	;
		938	
		939	AFCm:
033F	7642	940	jf1 ZrusPrg
0341	83	941	ret
		942	ZrusPrg: ;pri stisku AFC- nebo
0342	C5	943	sel rb0 ;pri pretecení volby zhasni display
0343	8978	944	orl p1,#Tma
0345	239F	945	mov a,#not (PrgCas or CasNapr)
0347	04BD	946	JMP anlSS ;a ve stavovem slove vynuluj bity,
		947	; ;signalizujici programovaci rezim
		948	;
		949	
		950	AFCp:
0349	764C	951	jf1 InitPrg
034B	83	952	ret
		953	InitPrg:
034C	FF	954	mov a,r7 ;vem stavove slovo
034D	323D	955	jb1 KonProg ;pri opakovane instrukci nebo kdyz je
034F	D23D	956	jb6 KonProg ;uz cas naprogramovan, nedelej nic
0351	B260	957	jb5 Tst0 ;programovalo-li se uz, testuj, zda
0353	2320	958	mov a,#PrgCas ;zadana hodnota <> 0,
0355	14B9	959	call orlSS ;jinak si poznamenej, ze se programuje
0357	D5	960	sel rb1 ;cas,

LOC	OBJ	LINE	SOURCE STATEMENT
0358	27	961	clr a ;vynuluj citac minut
0359	AB	962	mov r3,a
035A	AC	963	mov r4,a
035B	2380	964	mov a,#B7
035D	39	965	outl pl,a ;a zobraz .0
035E	643D	966	jmp KonProg
		967	Tst0:
0360	D5	968	sel rbl
0361	FB	969	mov a,r3
0362	4C	970	orl a,r4 ;nebyl jeste naprogramovan zadny cas,
0363	C63D	971	jz KonProg ;ignoruj stisk tlacitka, jinak proved
0365	5410	972	call Mult4 ;zaverecne nasobeni 4x (dva citace
0367	F642	973	jc ZrusPrg ;nameri jen 15 sekund a my merime
0369	27	974	clr a ;v minutach) a vynuluj zbyvajici dva
036A	A9	975	mov r1,a ;citace, aby se merilo uplne presne
036B	AA	976	mov r2,a
036C	8978	977	orl pl,#fma
036E	C5	978	sel rb0 ;pak zase zhasni display
036F	35	979	dis tcnt1
0370	2360	980	mov a,#PrgCas or CasNapr
0372	DF	981	xrl a,r7 ;a do stavoveho slova si poznamenej,
0373	04BF	982	JMP ZmenSS ;ze cas je uz naprogramovan
		983	; ret
		984	;
		985	-----
		986	TESTuC: ;test mikropocitace
0375	15	987	dis i
0376	35	988	dis tcnt1
0377	2350	989	mov a,#ChrA ;pri testu se pro kontrolu prubehu
0379	39	990	outl pl,a ;zobrazuje znak 'A'
		991	
037A	C68B	992	jz FatErr ;pokud nam to dovoli velikost EPROM,
037C	27	993	clr a ;doplnime sem jeste nekolik testu
037D	968B	994	jnz FatErr ;procesoru
037F	97	995	clr c
0380	F68B	996	jc FatErr
0382	A7	997	cpl c
0383	E68B	998	jnc FatErr
0385	A5	999	clr f1
0386	768B	1000	jf1 FatErr
0388	B5	1001	cpl f1
0389	768D	1002	jf1 Flok
		1003	FatErr: ;pri chybe mikropocitace
		1004	CRCerr: ;nebo pri vadne ROM se neda
038B	648B	1005	jmp \$;delat vubec nic
		1006	;
		1007	-----
		1008	Flok:
038D	B83F	1009	mov r0,#63 ;jeste otestujeme RAM
		1010	TestRAM: ;tak, aby se RAM nezmenila
038F	F0	1011	mov a,@r0
0390	37	1012	cpl a
0391	A0	1013	mov @r0,a
0392	D0	1014	xrl a,@r0
0393	968B	1015	jnz FatErr
0395	F0	1016	mov a,@r0
0396	37	1017	cpl a
0397	A0	1018	mov @r0,a
0398	B88F	1019	djnz r0,TestRAM
		1020	;
		1021	-----
		1022	;Test pameti programu. EPROM je zabezpecena CRCem s generujicim
		1023	;polynomem $x^{16} + x^{15} + x^2 + 1$. Pri priprave se CRC pocita
		1024	;ze vseh slabik programu krome dvou poslednich, vysledek se

LOC	OBJ	LINE	SOURCE STATEMENT
		1025	;pak ulozi na posledni dve adresy EPROM, pri kontrole se pocita
		1026	;se vsami adresami a vysledek musi byt nulovy. Vypocet CRC lze
		1027	;provest bud zvlastnim programem podle procedury CRC (viz nize)
		1028	;nebo pomoci emulatoru. Postup s pouzitim emulatoru je nasledujici:
		1029	; - preneseme pripraveny program do emulatoru
		1030	; - nastavime break-point na adresu 3FE
		1031	; - spustime beh programu se zastavenim na break-pointu
		1032	; - po zastaveni na adrese 3FE obsahuji registry R2 a R3 vypocitany
		1033	; CRC; obsah R2 ulozime na adresu 3FE, obsah R3 na adresu 3FF,
		1034	; tim je priprava hotova.
		1035	TestROM:
039A	BD50	1036	mov r5,#ChrA ;pri testu se pro kontrolu prubehu
039C	27	1037	clr a ;zobrazuji znaky 'B', 'C', 'D', 'E'.
039D	AA	1038	mov r2,a
039E	AB	1039	mov r3,a ;inicializace CRC-H, CRC-L,
039F	AE	1040	mov r6,a ;citace byte ve strance
03A0	A9	1041	mov r1,a ;a citace testovanych stranek
		1042	NxtPg:
03A1	FD	1043	mov a,r5 ;pri vstupu na nove stranky
03A2	0308	1044	add a,#8
03A4	AD	1045	mov r5,a ;zobraz jej "cislo"
03A5	39	1046	outl pl,a
		1047	NxtBy:
03A6	54E0	1048	call GetByte ;vem jeden byte
		1049	CRC: ;Vypocet CRC pro jeden byte
03A8	DA	1050	xrl a,r2
03A9	AA	1051	mov r2,a ;secti slabiku dat s CRC_H
03AA	BC08	1052	mov r4,#8 ;citac bitu
		1053	NxBit:
03AC	FB	1054	mov a,r3 ;CRC_L posun vlevo
03AD	6B	1055	add a,r3 ;do CY, zprava nula
03AE	AB	1056	mov r3,a
03AF	FA	1057	mov a,r2 ;CRC_H rotuj vlevo
03B0	7A	1058	addc a,r2 ;pres CY
03B1	AA	1059	mov r2,a ;predem uloz CRC_H
03B2	E6BB	1060	jnc CY0
03B4	D380	1061	xrl a,#80h ;kdyz CY = 1,
03B6	AA	1062	mov r2,a ;pricti jeste 80h k CRC_H
03B7	FB	1063	mov a,r3
03B8	D305	1064	xrl a,#05h ;a 05h k CRC_L
03BA	AB	1065	mov r3,a
		1066	CY0:
03BB	ECAC	1067	djnz r4,NxBit ;zpracuj pro vsech 8 bitu ve slabice
		1068	
03BD	1E	1069	inc r6 ;to proved pro vsechny byte ve strance
03BE	FE	1070	mov a,r6
03BF	96A6	1071	jnz NxtBy
03C1	19	1072	inc r1 ;a pro vsechny stranky
03C2	F9	1073	mov a,r1
03C3	37	1074	cpl a ;v 1 KB jsou jen 4 stranky
03C4	52A1	1075	jb2 NxtPg ;secti CRC-H a CRC-L, pokud nevysla 0,
03C6	FA	1076	mov a,r2 ;je vadna ROM a nemuzeme
03C7	4B	1077	orl a,r3 ;pokracovat
03C8	968B	1078	jnz CRCerr
03CA	8978	1079	orl pl,#Tma
03CC	04C6	1080	jmp Start ;na zaver jeste zhasni display
		1081	
		1082	;-
		1083	
		1084	DekodTbl:
		1085	
03CE	0A	1086	db 10 ;prekodovani instrukce; z 64 mozných
03CF	0D	1087	db 13 ;zpracovavame jen nektere
03D0	0E	1088	db 14 ;instrukce #0 : normovani analogu
			;instrukce #1 : umlceni, zapnuti
			;instrukce #2 : vypnuti

LOC	OBJ	LINE	SOURCE STATEMENT
03D1	01	1089	db 01 ;instrukce #3 : co kdo dava ?
03D2	00	1090	db 00 ;instrukce #4
03D3	00	1091	db 00 ;instrukce #5
03D4	00	1092	db 00 ;instrukce #6
03D5	00	1093	db 00 ;instrukce #7
03D6	00	1094	db 00 ;instrukce #8
03D7	00	1095	db 00 ;instrukce #9
03D8	00	1096	db 00 ;instrukce #10
03D9	00	1097	db 00 ;instrukce #11
03DA	00	1098	db 00 ;instrukce #12
03DB	00	1099	db 00 ;instrukce #13
03DC	00	1100	db 00 ;instrukce #14
03DD	00	1101	db 00 ;instrukce #15
03DE	00	1102	db 00 ;instrukce #16
03DF	02	1103	db 02 ;instrukce #17 : zapnuti, program 1
03E0	03	1104	db 03 ;instrukce #18 : zapnuti, program 2
03E1	04	1105	db 04 ;instrukce #19 : zapnuti, program 3
03E2	05	1106	db 05 ;instrukce #20 : zapnuti, program 4
03E3	06	1107	db 06 ;instrukce #21 : zapnuti, program 5
03E4	07	1108	db 07 ;instrukce #22 : zapnuti, program 6
03E5	08	1109	db 08 ;instrukce #23 : zapnuti, program 7
03E6	09	1110	db 09 ;instrukce #24 : zapnuti, program 8
03E7	00	1111	db 00 ;instrukce #25
03E8	00	1112	db 00 ;instrukce #26
03E9	00	1113	db 00 ;instrukce #27
03EA	00	1114	db 00 ;instrukce #28
03EB	00	1115	db 00 ;instrukce #29
03EC	00	1116	db 00 ;instrukce #30
03ED	00	1117	db 00 ;instrukce #31
03EE	00	1118	db 00 ;instrukce #32
03EF	00	1119	db 00 ;instrukce #33
03F0	00	1120	db 00 ;instrukce #34
03F1	00	1121	db 00 ;instrukce #35
03F2	00	1122	db 00 ;instrukce #36
03F3	00	1123	db 00 ;instrukce #37
03F4	00	1124	db 00 ;instrukce #38
03F5	00	1125	db 00 ;instrukce #39
03F6	0F	1126	db 15 ;instrukce #40 : hlasitost +
03F7	10	1127	db 16 ;instrukce #41 : hlasitost -
03F8	11	1128	db 17 ;instrukce #42 : jas +
03F9	12	1129	db 18 ;instrukce #43 : jas -
03FA	13	1130	db 19 ;instrukce #44 : sytost +
03FB	14	1131	db 20 ;instrukce #45 : sytost -
03FC	0B	1132	db 11 ;instrukce #46 : AFC +
03FD	0C	1133	db 12 ;instrukce #47 : AFC -
		1134	;dalsi instrukce uz nejsou zajimave
		1135	;a nemohou ani prijít
= 0030		1136	DelTab equ \$ - DekodTbl ;toto je delka tabulky
		1137	;*****
		1138	
03FE		1139	org 3FEH ;na konci pameti programu jsou data
		1140	DefCRC: ;pro její zabezpečení (tady se pro ne
03FE	DC	1141	db 0DCH ;symbolicky vymezí místo, CRC si
03FF	1D	1142	db 01DH ;samozřejmě vypočítáme a doplníme
		1143	;pomocným programem nebo emulátorem)
0000		1144	end

ASSEMBLY COMPLETE, NO ERROR(S)

USER SYMBOLS

?PROG	C 016B	?STOP	C 0186	?ZRUS	C 027B	ADD12	C 0209
APCM	C 033F	AFCMIN	C 01B4	AFCP	C 0349	AFCPLU	C 01B2
AKTHJS	D 0018	ANLSS	C 00BD	B0	= 0001	B1	= 0002

LOC	OBJ	LINE	SOURCE STATEMENT
B2	= 0004	B3	= 0008
B6	= 0040	B7	= 0080
CAS	= 0003	CASNAP	= 0040
CHRA	= 0050	CITACE	C 006F
CRCERR	C 038B	CY0	C 03BB
DACSYT	= 0040	DACY	C 0015
DEFAUL	C 021C	DEFCRC	C 03FB
DPLH	= 0013	DPLJ	= 0020
DKLICE	= 0005	EXTINT	C 0003
GETBYT	C 02B0	GTBYP0	C 02B8
GTBYP3	C 02F1	HJS1	D 0020
HJS4	D 0029	HJS5	D 002C
HJS8	D 0035	HJSMAX	C 01CE
HLASIT	C 0025	HLEDEJ	C 030A
HPLUS	C 019A	INITHJ	C 00B8
INSTBL	C 0149	IBET	C 007F
JAS1	C 0044	JDEBIT	C 02B6
JMPP1	C 01E0	JMPP2	C 02BE
KANAL?	D 0039	KANALA	D 0038
KLICRO	= 03FB	KNITEJ	C 0023
LASTCH	= 0005	LASTIN	D 003F
MOCHIT	C 02D1	MOZNO	C 0317
MULT4	C 0210	NAPROG	C 01B6
NEVHJS	C 001B	NEXTCH	C 00D5
NXBIT?	C 02CA	NXTBY	C 03A6
NXTCHN	C 0182	NXTPG	C 03A1
OVERFLW	C 021B	POCKEJ	C 0035
PRGCAS	= 0020	PRIJEM	C 0119
R00	D 0000	R01	D 0001
R04	D 0004	R05	D 0005
R11	D 0019	R16	D 001E
RETURN	C 007D	RGHLAS	D 001A
SMINUS	C 01D1	SNIZHJ	C 01D6
START	C 00C6	STARTB	C 02A2
STISK?	= 0004	STTLAC	D 001D
SYT0	C 0056	SYT1	C 0051
TD0	= 0015	TD0TD1	= 0019
TECKA0	C 00B5	TECKA1	C 00B1
TESTRA	C 038F	TBSTRO	C 039A
TNA	= 0078	TS	= 002B
TSTZLO	C 0140	UTLUM	C 002D
VYPINA	= 0080	VYPNUT	C 015E
ZAPNIT	C 01FF	ZAPPRG	C 01F5
ZMERNAT	= 0010	ZMRESS	C 00BF
ZRNEHI	C 005B	ZRNITO	= 0008
ZVYSHJ	C 01C4		
B4	= 0010	B5	= 0020
BEH	C 010B	BLIKRJ	C 00AD
CEKEJI	C 0226	CELESL	C 02D7
CNTINT	C 0007	CRC	C 03A8
DACHL	= 0010	DACJAS	= 0020
DALSIB	C 00F9	DALSIP	C 0244
DEKODT	C 03CB	DELTAB	= 0030
DPLS	= 0020	DPLTJS	C 021E
F1OK	C 038D	FATERR	C 038B
GTBYP1	C 02BE	GTBYP2	C 02BE
HJS2	D 0023	HJS3	D 0026
HJS6	D 002F	HJS7	D 0032
HLAS0	C 0030	HLAS1	C 0028
HMINUS	C 01A1	HOTOVO	C 0310
INITPR	C 034C	INITRA	C 00B4
JAS	C 0041	JAS0	C 0049
JMINUS	C 01B9	JMPP0	C 00C4
JMPTBL	C 02B4	JPLUS	C 01B3
KANALO	C 02F4	KLICRA	D 003A
KONPRO	C 033D	KONTRO	C 00F2
MAXHJS	= 0040	NEZICA	C 022F
MS200	= 0004	MULT2	C 0214
NAVRAT	C 007B	NESYNC	C 028C
NORMOV	C 018F	NXBIT	C 03AC
NXTCH1	C 0258	NXTCH2	C 0256
OPAKIN	= 0002	ORLSS	C 00B9
POHOTO	C 0111	PREPIN	= 0052
PROGRA	C 0312	PROVRD	C 0145
R02	D 0002	R03	D 0003
R06	D 0006	R07	D 0007
R17	D 001F	RAMOK	C 00EE
RGJAS	D 001E	RGSYT	D 001C
SPISLA	C 0067	SPLUS	C 01BF
STBIT0	C 0120	STBIT1	C 011F
SYNCHR	C 0281	SYNPUL	= 0008
SYTOST	C 004E	TD	= 0004
TD1	= 001D	TECKA	C 0080
TECKA0	C 003D	TESTKL	C 00CE
TESTUC	C 0375	TLACIT	C 0305
TST0	C 0360	TSTCAS	C 008A
UTLUMZ	C 01E8	VYPDO	= 0001
WBIT	C 02AD	ZADNAI	C 01DF
ZLON	= 000E	ZMENA	C 0236
ZPRACO	C 0138	ZPRTLA	C 0263
ZRUSPR	C 0342	ZRUSUT	C 01A8

AUTOMATIZACE PRŮMYSLOVÝCH A LABORATORNÍCH PROCESŮ POMOCÍ PERSONÁLNÍCH POČÍTAČŮ

Ing. Miloš Schlegel, CSc.

Článek se zabývá novými možnostmi v automatizaci průmyslových a laboratorních procesů, které přináší rozvoj (průmyslových) personálních počítačů (PC) a jejich příslušenství. Pozornost je věnována jak technickým prostředkům, tak aplikačnímu programovému vybavení (zahraničnímu i domácímu), které přináší pro uživatele nové, dříve netušené prostředky především v oblasti měření a zpracování signálů a v oblasti řízení technologických procesů.

1. Úvod

Personální počítače (PC), původně určené především pro kancelářské účely, se v současné době staly mocným prostředkem pro průmyslovou a laboratorní automatizaci a pro výzkum. PC vybavené přesnými vstupně/výstupními (V/V) analogovými a digitálními jednotkami mohou realizovat všechny funkce malých až středně rozsáhlých měřicích a řídicích systémů za zlomek jejich dřívější ceny; jsou k dispozici jak rozsáhlé stavebnice V/V jednotek a průmyslových PC, tak zpracované aplikační programové vybavení umožňující uživateli snadno, rychle a pružně konfigurovat potřebné funkce systému. PC s příslušným programovým vybavením nabízí standardní prostředky především v následujících oblastech průmyslu a výzkumu [1]:

- sběr dat, měření, zpracování signálů,
- řízení kvality výroby, testování,
- monitorovací a řídicí systémy technologických procesů.

Využitím těchto levných prostředků uživatel ušetří mnoho času, dříve spotřebovaného na vývoj vlastního jednoúčelového systému. Například doba potřebná pro instalaci malého řídicího a monitorovacího systému (desítky vstupů/výstupů) se tak zkrátí z měsíců (roků) na dny (týdny). Podstatné přitom je, že u uživatele se nepředpokládá znalost programování - specifikace funkcí systému se provádí vyplňováním formulářů nebo dokonce tvorbou grafických schémat. V současné době jsou pro řídicí systémy na bázi PC úspěšně vyřešeny i otázky spolehlivosti, ošetření činnosti při výpadku napájení, zálohování atd.

Cílem tohoto článku je

1) prezentovat základní technické parametry některých V/V jednotek firmy Advantech Corporation (desky PC-LabCard), o kterých lze předpokládat,

že naleznou uplatnění v našich podmínkách;

2) uvést stručnou charakteristiku zahraničního a domácího aplikačního programového vybavení spolupracujícího s deskami PC-LabCard;

3) podrobněji popsat vlastnosti nových programů Spurt a ProConT, které nabízí domácí firma Easy Control.

2. Připojení PC na proces

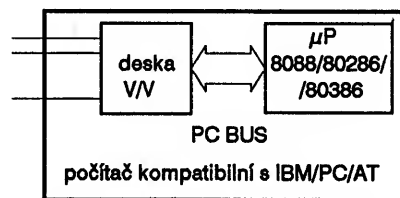
Existují dva základní způsoby připojení PC s měřicími a řídicími jednotkami:

a) přímé připojení jednotek na sběrnici PC;

b) připojení jednotek prostřednictvím standardních komunikačních kanálů RS-232, RS-242, RS-485 nebo IEEE-488.

Oba způsoby mají své přednosti a své nedostatky. Hlavní výhody přímého připojení jednotek na sběrnici PC (obr. 1) jsou následující:

- velká rychlost přenosu,
- nízká cena,
- malé rozměry.

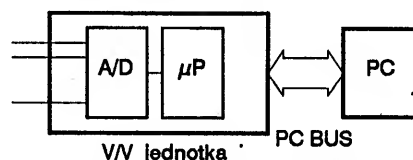


Obr. 1. Přímé připojení V/V jednotky na sběrnici PC

Velká rychlost je umožněna vyloučením relativně pomalého komunikačního protokolu. Nízkou cenu a malé rozměry způsobuje skutečnost, že není třeba další napájecí zdroj a ochranná skříňka, neboť V/V desky se zasouvají přímo do sběrnice konektoru (PC BUS) uvnitř PC. Maximální počet už-

tých desek pro jedno PC nemůže být tedy větší než počet volných pozic (slotů) uvnitř počítače. V současné době existuje mnoho úspěšných výrobců desek tohoto typu. Uvedme alespoň nejznámější - Data Translation, Metrabyte, Analog Devices, Burr-Brown, Advantech Co.

Výhody dosažené užitím standardních komunikačních protokolů (obr. 2) jsou následující: rozsah systému (počet vstupů/výstupů) není prakticky omezen, V/V jednotky mohou být umístěny přímo v blízkosti zdrojů signálů (minimalizace kabelového spojení, eliminace rušení, ...), V/V jednotky mohou být připojeny k libovolnému typu počítače pomocí standardního programového vybavení.



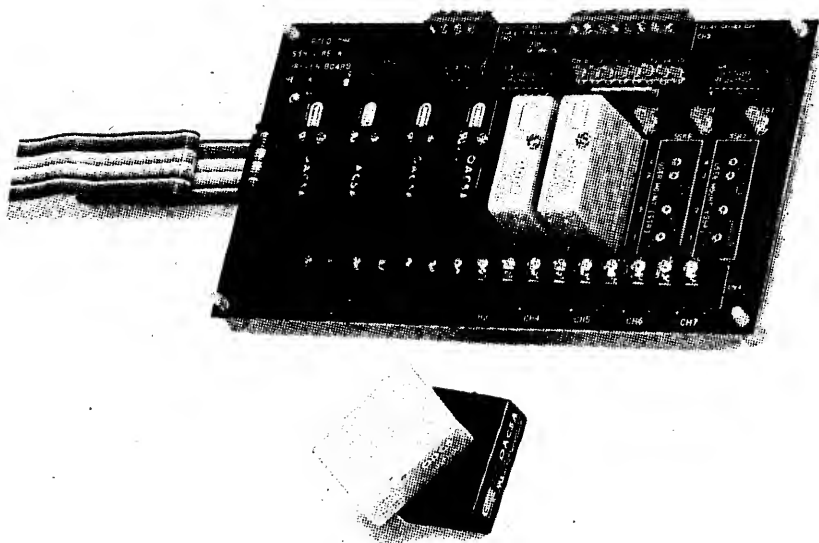
Obr. 2. Připojení V/V jednotky standardním komunikačním kanálem

Komunikace prostřednictvím RS-232, RS-242, RS-485 nebo IEEE-488 vyžaduje, aby V/V jednotky měly svůj vlastní mikroprocesor, který řídí přenos dat a navíc může provádět lokální funkce a tím odlehčit hostitelskému počítači.

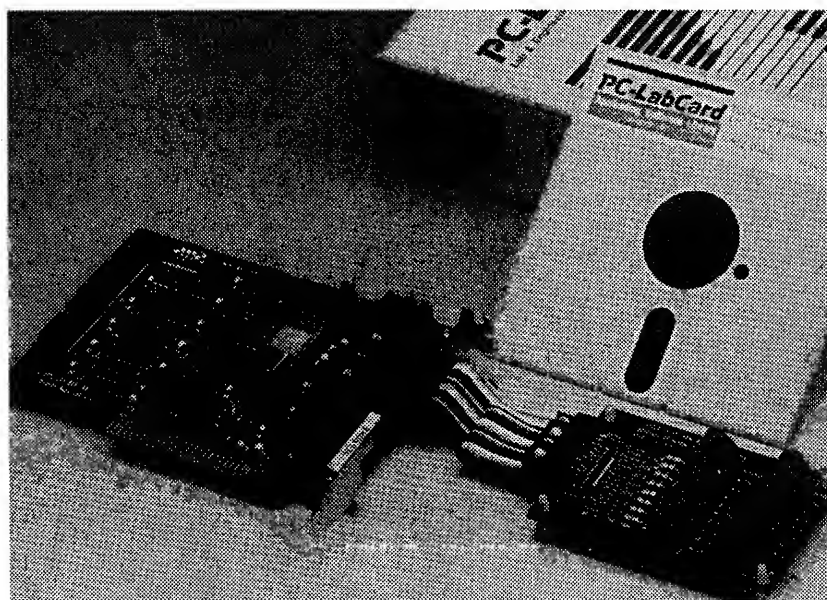
Nejznámější výrobci V/V jednotek připojitelných přes standardní komunikační kanály jsou Hewlett-Packard, Fluke, Keithley.

3. Stavebnice PC-LabCard

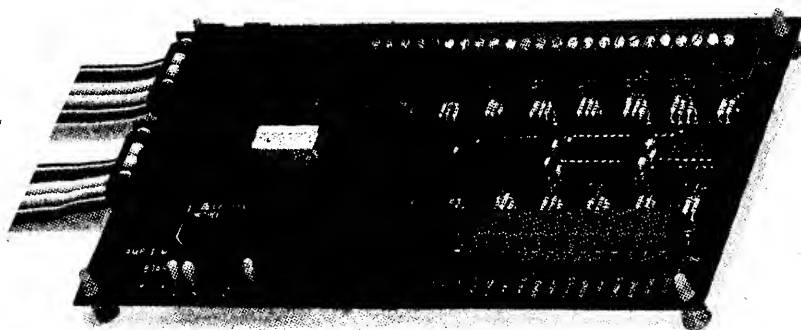
Firma Advantech Co. vyrábí ucelenou řadu technických prostředků (PC-LabCard Series) umožňujících komplexní řešení průmyslové a laboratorní automatizace. Podle rozsahu a charakteru aplikace lze vybrat od jednodu-



Obr. 3. PC-LabCard PCLD-786, osm opticky oddělených spínačů



Obr. 4. PC-LabCard PC-711C, nejlevnější souprava univerzálního použití



Obr. 5. PC-LabCard PCLD-789, multiplexer a zesilovač

chých V/V desek až po průmyslové PC vhodné do náročných prostředí. V tabulce 1 je uveden (neúplný) přehled V/V desek a jejich technických parametrů. Desky PC-LabCard se vyznačují tím, že obvykle plní několik funkcí současně. Například nejjednodušší a nejlevnější deska PCL-711S obsahuje 8 jednopólově uzeměných analogových vstupů, 1 analogový výstup, 16 digitálních vstupů a 16 digitálních výstupů. Tato skladba vstupů a výstupů umožňuje v jednoduchých případech vystačit pouze s jedinou V/V deskou. Další charakteristická vlastnost desek PC-LabCard je konfigurace funkcí buď pomocí přepínačů umístěných na desce nebo programovými prostředky. Například vstupní rozsah analogových vstupů na desce PCL-818 lze volit přepínačem nebo programově (bipolární rozsahy $\pm 10, 5, 2, 2, 5, 1, 0, 5$ V; unipolární rozsahy 0 až 10, 5, 2, 1 V). Veiká pružnost stavebnice PC-LabCard je však dosažena především tím, že kromě desek vkládaných do PC má uživatel k dispozici široký sortiment tzv. dceřiných desek, umístovaných mimo vlastní PC - tím je podstatným způsobem rozšířena aplikovatelnost stavebnice. K libovolné desce uvedené v tabulce 1 a k některým dalším lze připojit (alternativně) následující dceřiné desky:

- PCLD-782: 16 izolovaných digitálních vstupů.
- PCLD-785: 16 releových výstupů (120V/1A).
- PCLD-786: 8 opticky oddělených výkonových spínačů AC (280V/3A) nebo DC (60V/3A).
- PCLD-788: releový multiplikátor (16 na 1).
- PCLD-770: multiplexer (8 na 1) a přizpůsobení analogových signálů s možností optického oddělení, zesílení, kompenzaci studeného konce pro termočlánky, ...
- PCLD-789: multiplexer (16 na 1) a zesilovač (až 1000x).
- PCLD-889: multiplexer (16 na 1), zesilovač s programovatelným zesílením (až 1000x).

Některé dceřinné desky lze dále řadit do kaskády. Tímto způsobem můžeme například vytvořit multiplexer přepínající až 256 kanálů na jeden kanál.

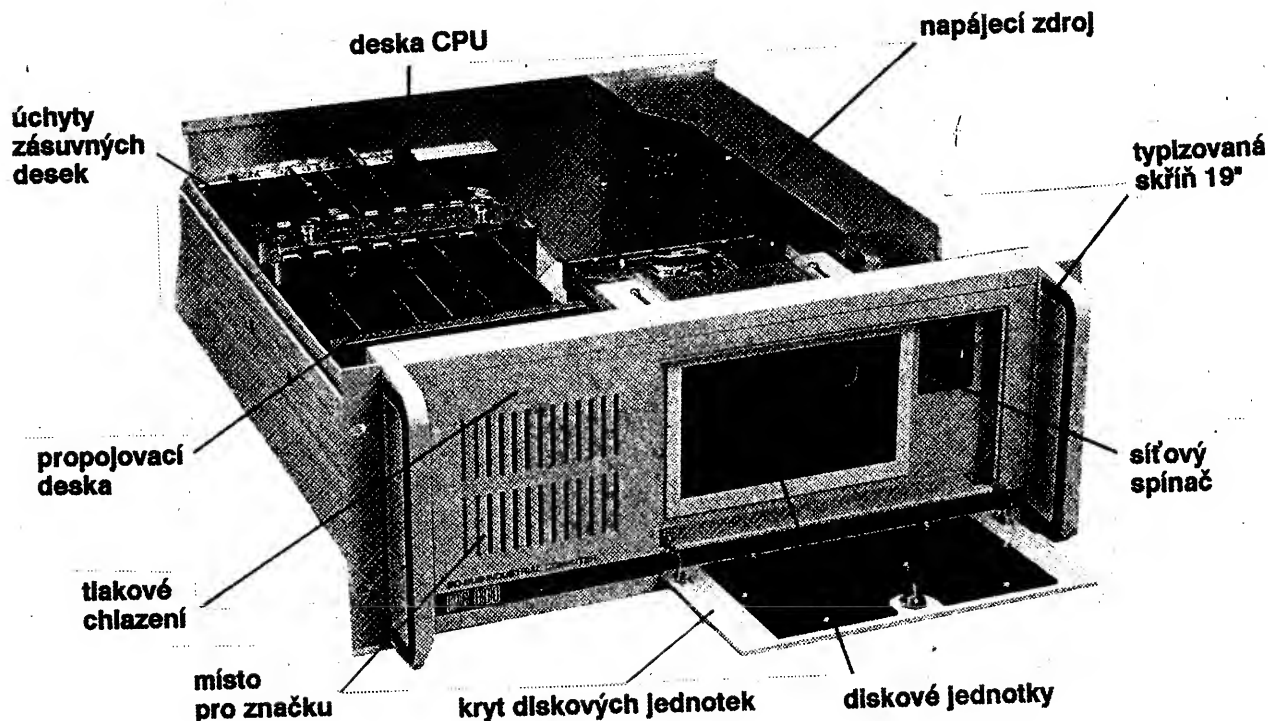
Desky PC-LabCard umožňují i připojení PC k různým zařízením přes standardní komunikační kanály. Deska PCL-743 zajišťuje sériovou komunikaci (RS422, RS-485) a deska PCL-848 A/B umožňuje paralelní komunikaci (IEEE-488).

Desky PC-LabCard dodává v ČSFR např. firma FCC Folprecht, s. r. o., Velká hradební 48, 400 01 Ústí nad Labem, tel. (047) 26308.

Všechny desky PC-LabCard mohou spolupracovat s libovolným počítačem.

	PCL-711S	PCL-812	PCL-812PG	PCL-718	PCL-818	PCL-714	PCL-880	PCL-720	PCL-722	PCL-726	PCL-830
A-D kanály	12 bitů 8 jednopól.	12 bitů 16 jednopól.	12 bitů 16 jednopól.	12 bitů 16 jednopól. nebo 8 dif.	12 bitů 16 jednopól. nebo 8 dif.	14 bitů 16 dif.	4,5 číslice 1 izol. kanál				
Vstupní rozsahy [V]	±5	±10, 5, 2, 1 volba přep.	±10, 5, 2, 5, 1,25, 0,625, 0,312 volba prog.	±10, 5, 2, 5, 1, 0,5, 0 až 10, 5, 2, 1 volba přep.	±10, 5, 2, 5, 1, 0,5, 0 až 10, 5, 2, 1 -volba přep. i prog.	±5, 1 V volba přep.	±200, 20, 2, 0,2				
Vzorkovací kmitočet [kHz], čas konverze [μs]	25 25	30 25	30 25	60 nebo 100 12 nebo 8	100 8	20 40	2+10 Rdgs/s				
Hardwarové přepínání kanálů	ne	ne	ne	ano	ano	ano	ne				
Spouštění převodu prog. časovačem	ne	ano	ano	ano	ano	ano	ne				
Přenos dat DMA	ne	ano	ano	ano	ano	ano	ne				
D/A kanály	12 bitů 1 kanál	12 bitů 2 kanály	12 bitů 2 kanály	12 bitů 2 kanály	12 bitů 2 kanály	14 bitů 2 kanály				6 kanálů	
Výstupní rozsahy [V]	0 až 5 0 až 10	0 až 5	0 až 5 0 až 10	0 až 5	0 až 5 0 až 10	0 až 5				±5, 10, 0 až 5, 10, 4 až 20 mA	
Digitální V/V TTL	16 vst. 16 výst.	16 vst. 16 výst.	16 vst. 16 výst.	16 vst. 16 výst.	16 vst. 16 výst.	16 vst. 16 výst.	16 výst.	32 vst. 32 výst.	144 vst. 144 výst.	16 vst. 16 výst.	16 vst. 16 výst.
Programovatelný čítač	ne	1 kanál (8253)	1 kanál (8253)	1 kanál (8254)	1 kanál (8254)	1 kanál (8253)		3 kanály (8253)			10 kanálů (ADM9513 x2)
Cena (USD)	295	395	450	845	995		550	160	350	495	325
Základní program. podpora	BASIC/C/ Pascal	BASIC/C/ Pascal	BASIC/C/ Pascal	BASIC/C/ Pascal	BASIC/C/ Pascal	BASIC/C/ Pascal	BASIC/C/ Pascal				BASIC/C/ Pascal
Zahrnutí aplikační programy	PC-LabDAS LT-NoteBook LT-Control PC-Scope Control-EG UnkelScope	PC-LabDAS LT-NoteBook LT-Control PC-Scope Control-EG UnkelScope SnapsShot	PC-LabDAS LT-NoteBook LT-Control PC-Scope Control-EG UnkelScope SnapsShot	PC-LabDAS LT-NoteBook LT-Control PC-Scope Control-EG UnkelScope SnapsShot ASYST ASYSTANT +	PC-LabDAS LT-NoteBook LT-Control PC-Scope Control-EG UnkelScope SnapsShot ASYST ASYSTANT +	PC-LabDAS UnkelScope	PC-LabDAS	PC-LabDAS Control-EG		PC-LabDAS Control-EG	
Československé aplikační programy	EC-Expert ProCont	EC-Expert Spurt ProCont	EC-Expert Spurt ProCont	EC-Expert Spurt ProCont	EC-Expert Spurt ProCont	EC-Expert Spurt ProCont		ProCont	ProCont	ProCont	

Tab. 1. Přehled vybraných desek V/V PC-LabCard (deska PC-714 je velmi rozšířená, už se však nevyrábí ani nedodává).



Obr. 6. Průmyslový počítač IPC-600

tačem kompatibilním s IBM PC/XT /AT/386. Do náročného průmyslového prostředí je však vhodné místo „obyčejného PC“ použít průmyslové provedení PC, které má následující přednosti:

- skříňka PC je vybavena 2 ventilátory (s výměnným filtrem), které vytvářejí uvnitř skřínky přetlak a tím zabraňují vnikání prachu a dalších nečistot,
- desky uvnitř PC jsou zafixovány speciálním antivibračním držákem,
- diskové jednotky jsou uchyceny pomocí pryžových bloků, které tlumí vibrace,
- disketové jednotky jsou opatřeny speciálním krytem proti vnikání nečistot,
- pohyblivé disky mohou být nahrazeny tzv. RAM/ROM diskem,
- průmyslový PC může pracovat bez klávesnice, popř. i bez monitoru (tj. bez zásahů operátora),
- zaručené provozní podmínky: teplota 0-50°C; vlhkost 10- 85% při 40°C; vibrace 5-17 Hz při rozkmitu 0,2", 17-20 Hz při 1G (špička-špička), 200-500 Hz při 3G (špička-špička),
- zahoření systému: 72 hod. při teplotě 50°C.

Příklad konfigurace průmyslového PC s deskami V/V je uveden v Tab. 2.

Přehled aplikačního programového vybavení

Kvalitní využití potenciálních možností desek PC- LabCard je ve většině případů možno dosáhnout pouze zakoupením vhodného profesionálního

aplikačního programového vybavení. Jen těžko lze předpokládat, že by sám konečný uživatel byl ochoten vynaložit tolik času a úsilí pro vytvoření srovnatelného programového produktu. Obzvláště tehdy, existuje-li skutečně lákavá nabídka. Uživatel však může mít speciální požadavky, které nemusí být realizovatelné pomocí zabudovaných funkcí. V tomto případě je podstatné, zda příslušný programový balík připoustí rozšíření, a zda jej může provést jednoduše sám uživatel. Další důležitá kritéria pro výběr programů tohoto typu jsou:

- kvalita a množství zabudovaných funkcí,
- jednoduchost obsluhy (kvalita uživatelského prostředí),
- čas potřebný pro zaškolení,
- kvalita a úplnost uživatelské dokumentace,
- reference.

Mnoho informací o programu lze získat též shlednutím jeho demonstrační verze.

V tabulce 3 je uveden přehled programových balíků určených pro měření a zpracování signálů, testování, monitorování a řízení průmyslových a laboratorních procesů. Údaje uvedené v tabulce mají sloužit pouze pro hrubou orientaci. Podrobnější informace lze nalézt v [2], nebo přímo u dodavatele.

V tabulce 4 je přehledně uvedeno, které karty PC- LabCard jsou podporovány jednotlivými programy.

značení	popis	cena
IPC-600	skříňka průmyslového PC s 8 sloty	1350 \$
-opt. 220	pružný disk 5,25" 1,2 MB	115 \$
-opt. 233	pevný disk 40 MB	480 \$
PCA-6125/1M	CPU 80286, 1 MB RAM, diskové řadiče, sériové a paralelní rozhraní	510 \$
PCA-6300	standardní klávesnice PC AT	70 \$
PCA-6300FF	ochranná fólie na klávesnici	25 \$
PCA-6141	video karta VGA	190 \$
IPC-651M	barevný monitor Multisync 14"	580 \$
PCL-812	kombinovaná V/V deska	450 \$
PCL-718	kombinovaná V/V deska	845 \$
PCL-726	deska analogových výstupů	495 \$
PCL-722	dlg. deska V/V 144 bitová	350 \$
PCLD-788	reléový multiplexer 16 na 1	250 \$
celkem		5710 \$
Počty vstupů a výstupů:		
- 47 analogových vstupů,		
- 10 analogových výstupů,		
- až 192 digitálních vstupů,		
- až 192 digitálních výstupů.		

Tab. 2. Příklad konfigurace průmyslového PC

Název	Oblast aplikací	Poznámka	f_{max} kHz	V/V funkce	Cena US \$
PC-LabDAS	měření, monitorování a testování výroby	zdrojový program v QUICK-BASICu může být dodán k modifikaci	0,01	A/D, D/A, DIO	495
PC-SCOPE	emulace paměťového osciloskopu	velmi jednoduché užití		A/D	95
LT-NoteBook	měření a řízení laboratorních a průmyslových procesů	shora kompatibilní s LT-Control	12,5 až 100	A/D, D/A, DIO, IEEE-488, RS-232	995
LT-Control	monitorování a řízení procesů, distribuované řízení	může být dodán Run-Time systém (nakonfigurovaný)		A/D, D/A, DIO, RS-232	3395
Control-EG	jednoduché monitorování a řízení, emulace regulátorů PID a programovatelných automatů	Jednoduché užití	0,01	A/D, D/A, DIO, RS-232	500
ASYSTANT+	měření a jeho zpracování	řízení pomocí menu	30	A/D, D/A	995
ASYSTANT-GPIB	měření a jeho zpracování	jako ASYSTANT+, pro přístroje s IEEE-488		IEEE-488	695
ASYST	měření a jeho zpracování	mnoho výkonných funkcí pro zpracování signálů	60	A/D, D/A, DIO, IEEE-488, RS-232	2095
UnkelScope	měření a zpracování signálů, jednoduché řízení		100	A/D, D/A, DIO	549
DADISP	zpracování signálů	160 funkcí pro zpracování a generování signálů		pouze datové soubory	895
Spurt	měření a zpracování signálů pro průmyslové a výzkumné účely	jednotné uživatelské prostředí, makroinstrukce	60	A/D	Kčs 30tis.
ProConT	monitorování a řízení laboratorních a průmyslových procesů	řídící systém regulačního i logického typu, automatické nastavování průmyslových regulátorů	1	A/D, D/A, DIO	Kčs 20tis.
EC-Expert	automatické nastavování průmyslových regulátorů		1	A/D, D/A,	Kčs 18tis.

Tab. 3. Přehled programového vybavení podporujícího desky PC-LabCard

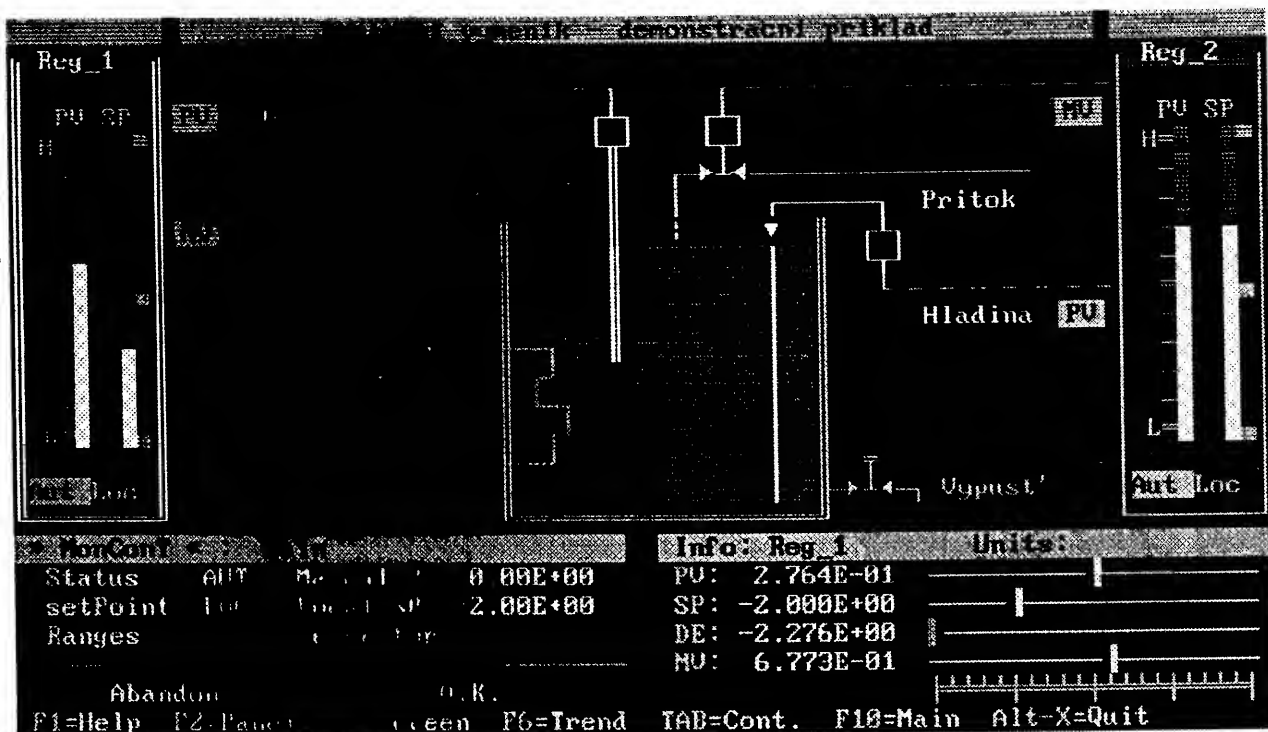
karta PCL-	860	711 S	812	812 PG	718	818	720	722	726	830	848	725
PC-LabDAS	*	o	o	o	o	o	o		o	*		o
PC-SCOPE		o	o	x	o	x						
LT-NoteBook		o	o	x	o	x	*	*	*	*	*	
LT-Control		o	o	x	o	x	*	*	*	*	*	
Control-EG		o	o	x	o	x	o		o			o
ASYST					o	x					o	
ASYSTANT+					o	x						
ASYSTANT-GPIB											o	
UnkelScope		o	o	x	o	x						
Spurt			o	x	o	x						
ProConT		o	o	x	o	x	o	o	o			*

Tab. 4. Programová podpora desek PC-LabCard
(o - plná podpora, x - částečná podpora, * - připravuje se)

5. Spurt - program pro měření a zpracování signálů

Spurt je ucelený programový systém pro počítače typu IBM PC/XT /AT/386 vyvinutý firmou Easy Control. Je použitelný v široké oblasti vědeckotechnických aplikací, v laboratorních i provozních podmínkách při měření a zpracování naměřených signálů. Program vytváří integrované uživatelské prostředí, ve kterém je dialog uživatele s počítačem veden formou hierarchických menu a speciálními grafickými prostředky. Uživatel i bez znalosti programování může v tomto prostředí řešit mnoho problémů z oblasti získávání a analýzy experimentálních dat. Program Spurt je otevřeným systémem. Jeho základní operace pro zpracování signálů může uživatel jednoduchým způsobem doplňovat (subsystém makroinstrukcí).

Typický postup uživatele při práci s programem je následující:



Obr. 7. Ukázka z obrazovky programu ProCont firmy Easy Control

1. **Konfigurace** - uživatel zadává výběr, umístění a nastavení měřicích desek a dále specifikuje tzv. experiment, tj. měřená místa, formu předzpracování signálů, zobrazování v reálném čase, způsob spouštění měření, periodu vzorkování a případné další atributy experimentu.

2. **Monitorování** - uživatel ověřuje, zda měřené signály jsou řádně napojeny na měřicí systém a zda měřené veličiny jsou v pořádku. Tento režim může být též použit pro dlouhodobé monitorování technologických veličin a výstražná hlášení.

3. **Měření** - uživatel provádí vlastní měření, a to buď dynamické nebo jednorázové. Během dynamického měření mohou být na obrazovce zobrazeny jako na osciloskopu až 4 měřené veličiny.

4. **Zpracování** - uživatel analyzuje naměřená data, přičemž má k dispozici celou řadu standardních prostředků pro zpracování signálů (aritmetické operace, spektrální analýza, číslicová filtrace, základní statistika, ...). Zpracovávaná data mohou být zobrazována jak v numerické, tak v grafické formě. Všechny zabudované funkce mohou být začleněny do uživatelských makroinstrukcí. Naměřené i zpracované hodnoty je možné vypsat na tiskárnu, nebo ploter. Navíc naměřená data z jednotlivých kanálů mohou být exportována ve formě ASCII souboru do jiných programů pro analýzu časových řad a naopak.

Poznámka: Měřená data jsou ukládána do operační paměti, z čehož plynou určitá omezení: např. pro počítač s operační pamětí 640 kB a 16 měře-

ných kanálů je maximální počet vzorků na jeden kanál 4096. Pro menší počet kanálů je možné úměrně zvětšit maximální počet vzorků (nemůže však v žádném případě překročit 32K vzorků). Minimální realizovatelná perioda vzorkování závisí na počtu měřených kanálů, na rychlosti počítače a příslušného A/D převodníku. Pro jeden kanál je dosažitelný vzorkovací kmitočet podle typu desky 20 až 80 kHz

Parametry programu Spurt

Konfigurace a měření

- možnost obsluhy až 8 měřicích desek,
- současné měření až 128 kanálů s volitelným rozsahem,
- převod na fyzikální jednotky, jednoduchá filtrace v reálném čase,
- různé způsoby spouštění experimentu,
- rychlost měření: 1 kanál až 80 kHz, 10 kanálů až 1 kHz,
- řízení procesu pomocí PID regulátorů a sekvenčních tabulek.

Analýza

- současné zpracování až 8 signálů,
- FFT signálů s maximálně 8192 vzorky,
- rozšiřitelnost vestavěných funkcí pomocí makroinstrukcí,
- grafické výstupy na obrazovku, tiskárnu a ploter,
- export a import dat pro jiné programy.

Modifikace uživatelského prostředí

- velikost obrázků, barvy, typy čar,
- změna počtu kanálů a počtu vzorků na kanál,
- informace o programu.

6. ProConT - programový balík pro řízení technologických procesů

ProConT je rodina programů (ExeConT, MonConT, ConConT, TunConT), která umožňuje snadno, rychle a kvalitně navrhnout a realizovat malý až středně rozsáhlý řídicí a monitorovací systém na bázi PC a V/V desek PC-LabCard (nebo podobných). Konfigurace takového systému nevyžaduje ani znalost programování, ani speciální zkušenosti z oblasti číslicového řízení.

Zabudované řídicí algoritmy a monitorovací funkce, které plně postačí pro většinu běžných aplikací, lze snadno rozšířit pomocí uživatelských programů v jazyce Turbo Pascal. ProConT tímto způsobem umožňuje nejen implementaci nestandardních algoritmů řízení, ale i simulaci ověření řídicího systému se speciálním matematickým modelem technologického procesu.

ProConT je velmi vhodný především pro řízení technologických uzlů (pěce, výměníky, sušárny, čisticí stanice, chemické a biologické procesy, ...), kde se vyskytují úlohy jak regulačního, tak logického typu. Rozsah aplikace je omezen především technickými parametry použitých V/V desek. Rozumně dosažitelné (při použití desek PC-LabCard) jsou následující počty:

- až 100 analogových vstupů,
- až 20 analogových výstupů,
- stovky digitálních vstupů/výstupů,
- až 64 regulačních smyček.

Typický postup při práci s programem ProConT je následující:

1. **Konfigurace (ConConT)** - uživatel konfiguruje (pomocí hierarchického systému menu) všechny žádané funkce řídicího a monitorovacího systému. Sem patří: konfigurace technických prostředků, vstupů/výstupů, řídicích algoritmů, monitorovacích snímků a exekutivy reálného času. Ve specifikaci řídicích algoritmů lze využívat následující zabudované prostředky: předpracování analogových signálů (číslíková filtrace, konverze na fyzikální jednotky, alarmy), standardní průmyslové regulátory, generátory časových funkcí pro zadávání požadovaných hodnot, sekvencí tabulky pro specifikaci sekvencí a kombinační logiky a alarmového subsystému. Vybrané technologické veličiny lze zařadit do krátkodobé nebo dlouhodobé archivace. Výsledkem konfigurace je „konfigurační databáze“, která plně definuje zkonfigurovaný řídicí systém. Na jejím základě lze automaticky vygenerovat úplnou dokumentaci řídicího systému.

2. **Instalace řídicího systému.** ProConT pracuje pod operačním systémem MS DOS (verze 3.10 a vyšší) v dvojúrovňovém prostředí Easy Control:

- na pozadí běží rezidentní exekutiva reálného času ExeConT, která spouští nakonfigurované řídicí a archivační funkce a zařazené uživatelské programy,
- na popředí typicky pracuje monitorovací (MonConT), nebo konfigurační (ConConT), popřípadě optimalizační (TunConT) program, může zde však být spuštěn libovolný program operačního systému MS

DOS (např. editor, databáze, tabulkový procesor, grafický editor, překladač, ...).

Ve fázi uvádění do provozu se na popředí obvykle střídavě spouští všechny tři programy:

- ConConT: změna konfigurace,
- MonConT: ověření správné činnosti,
- TunConT: automatické nastavování standardních průmyslových regulátorů (P,I,PI,PD,PID).

3. **Provozování řídicího systému.** Běžnou činnost řídicího systému zajišťují programy ExeConT a MonConT. V případě, že není nutné řízený proces neustále monitorovat, lze na popředí spustit jakýkoli jiný program podle potřeby uživatele. V takovém případě jsou všechny alarmy hlášeny akusticky. Při výskytu alarmu by měl operátor ukončit běžící program, spustit monitorovací program řídicího systému MonConT a v něm potom vhodným způsobem ošetřit výstražná hlášení.

Základní technické parametry programu ProConT

- maximální počet vstupů/výstupů je omezen pouze technickými prostředky (parametry V/V desek);
- maximální počet standardních regulátorů je omezen pamětí RAM: standardně lze nakonfigurovat až 64 regulátorů s volitelnými (obecně různými) periodami vzorkování;
- minimální perioda vzorkování PID regulátoru je menší než 5 ms (PC/XT 8MHz);
- výstupy regulátorů mohou být analogové (amplitudová modulace) nebo logické (šifrová modulace, on-off, třístavový výstup pro ovládání servopohonu);

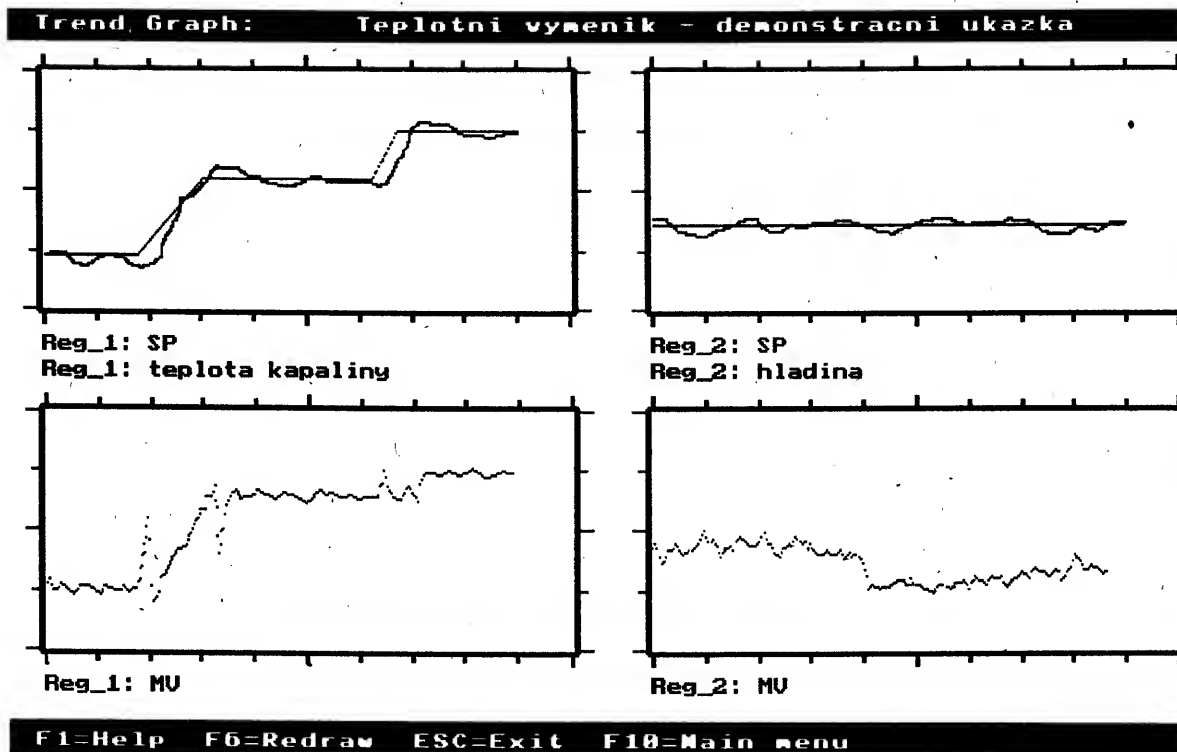
- maximální počet generátorů časových funkcí je omezen pamětí RAM: standardně lze nakonfigurovat až čtyři generátory typu „lomená čára“ s maximálně 16 úseky;
- maximální počet sekvenčních tabulek je omezen pamětí RAM;
- pod exekutivu reálného času lze zařadit až čtyři uživatelské programy a v každém z nich je možné naprogramovat (v jazyce Turbo Pascal na základě dodávané knihovny) až 8 úloh (vstupní/výstupní operace, řídicí algoritmy, ...);
- rezidentní část (ExeConT s řídicími algoritmy a konfigurační databází) zabírá přibližně 100 kB paměti.

7. Závěr

Personální počítače spolu s přídatnými V/V deskami a vhodným aplikačním programovým vybavením nabízejí prostředky pro jednoduchou, účinnou, spolehlivou a hlavně levnou automatizaci pro široký rozsah průmyslových a laboratorních procesů. Hlavní oblastí aplikací jsou měření a zpracování naměřených dat, testování výroby a řízení a monitorování malých a středně rozsáhlých technologických procesů. Ve všech těchto oblastech mají personální počítače v našich podmínkách velké možnosti.

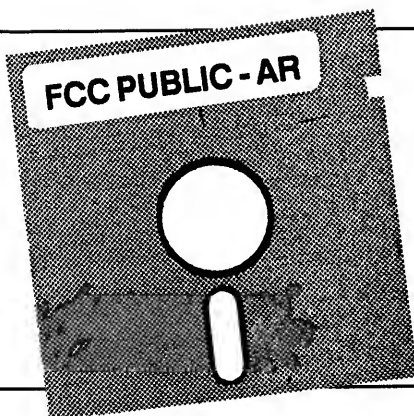
Literatura

- [1] Control Engineering, June 1990.
- [2] Katalog firmy Advantech Co., 1990.



Obr. 8. Ukázka grafického zpracování měřených hodnot programem ProConT

NABÍDKA
1500 PROGRAMŮ
PUBLIC DOMAIN
SOFTWARE



PUBLIC DOMAIN

VOLNĚ ŠÍŘENÉ PROGRAMY

VOLNĚ ŠÍŘENÉ PROGRAMY

PUBLIC DOMAIN SOFTWARE OD FCC PUBLIC

V krátkém úvodu se nejprve zamyslíme nad tím, co vlastně znamená slovní spojení **PUBLIC DOMAIN**, jaký je jeho význam a smysl. Potom vysvětlíme, jakými cestami se Public Domain po světě šíří a nakonec Vám podrobně přiblížíme již existující a tedy dostupné, ale i teprve chystané aktivity firmy FCC Folprecht spol. s r.o. v oblasti Public Domain software. Na dalších stránkách najdete seznam všech v současné době objednatelných programů a konečně podrobnější popis některých programů z prvních disket edice FCC Public.

Pod termínem **PUBLIC DOMAIN** software jsou myšleny programy, které se nerozšiřují komerčně - tedy prodejem. Vytvořili je lidé, kteří se jimi nehodlají obohacovat; naopak dali své výtvo-ry k dispozici ostatním, k veřejnému použití.

Tohle je jejich, pro někoho snad zvláštní chování, může být zapříčiněno různými pohnutkami: tvůrci mají třeba radost z programování a nepotřebují si tak vydělávat nebo chtějí ostatním ukázat své schopnosti, někteří možná k smrti neradi obchodují a tak své výtvo-ry raději nabízejí zdarma.

Ať tak či onak, Public Domain soft-ware je tu a my máme šanci jej využít.

Public Domain software nemá nic společného s „pirátským“ kopírováním programů nebo snad dokonce s krádeží, která by se dala kvalifikovat jako trestný čin. Právě naopak! Tvůrci těchto produktů dobrovolně dávají své pro-gramy k dispozici široké veřejnosti a jsou si přitom vědomi, že se jejich programy budou - jsou-li opravdu dob-ře - laťvinně šířit mezi uživateli.

Název Public Domain se však ob-čas používá příliš široce, často i na produkty, kterým toto označení roz-hodně nepatří. Můžeme rozlišovat cel-kem tři různé oblasti programových produktů, které bývají někdy souhrnně označeny jako Public Domain.

1. Public Domain

Takzvaný pravý Public Domain software označuje programy, které se mohou šířit a používat úplně volně, nezávisle na autorských právech. S těmi- to programy si můžete dělat prakticky co chcete: jejich části můžete např. po- užívat ve vašich vlastních programech, můžete je analyzovat, vylepšovat a tyto vylepšené verze sami dále rozšiřovat - a to všechno aniž byste žádali původ- ního autora o svolení nebo aniž byste ho jako původního autora jmenovali. V tomto posledním bodě by však měla vládnout jakási džentlimentská dohoda - že byste prostě neměli autorovi upírat jeho autorství - zvláště když on ani žád-

né finanční vyrovnání za použití pro- gramu nevyžaduje.

2. Freeware

Sem patří skupina těch programů, u nichž nelze (tak jako u pravého Pub- lic Domain) libovolně potlačit původní autorská práva. Tyto programy si mů- žete zdarma kopírovat a rozšiřovat při zachování těchto podmínek:

- a) všechny soubory, příslušející k nějakému programu, musí zůstat po- hromadě,
- b) autorství se nesmí upírat,
- c) nemůžete využívat pro své účely nějakou část takového software aniž byste přitom nepožádali autora o svolení.

V žádném případě nesmíte tyto pro- gramy dále používat k vlastnímu obo- hacení!

3. Shareware

Poslední skupinou je tzv. sharewa- re, což je vlastně komerční software, o jehož prodej se však tvůrci sami při- líš nechtějí starat. Tento druh programů se v říši osobních počítačů vyskytuje nejčastěji. S placením je to vymyšleno takto: můžete si zdarma třeba od své- ho přítele takový produkt nahrát, jeho tvůrce však očekává, že když pro- gram začnete používat, zašlete mu nějaký poplatek jako odměnu za jeho vyná- loženou námahu při tvorbě. Obsáhlé zkoušení tohoto software je Vám po- voleno, ale když se rozhodnete pro jeho nasazení na Váš problém, očeká- vá proste autor na základě svých autor- ských práv poplatek. Výše tohoto po- platku je nějakou vhodnou formou o- známena při spuštění či ukončení pro- gramu a většinou se pohybuje mezi 10 a 50 americkými dolary.

Po zaplacení poplatku obdrží uživa- tel registrační kartu a je s ním „počítá- no“, může například přednostně získat nejnovější verze produktu.

Pro příklad nemusíme daleko - všichni jistě známe a používáme ob- líbené „pakovače“ a „rozpakovávače“

americké firmy PKWARE Inc.; ale kdo z nás už zaslal požadovaný poplatek?

Při pravidelném používání tako- vých produktů je asi prostou lidskou slušností požadovaný registrační po- platek zaplatit.

PC-SIG Library

V roce 1982 byla v Kalifornii ve Spo- jených státech založena firma s ná- zvem PC-SIG Inc., která se začala za- bývat shromažďováním a distribucí shareware a později i Public Domain software. Distribuce probíhala na kla- sických disketách formátu 5,25" DD.

Edice této firmy s označením PC- SIG Library se za několik let rozrostla téměř neuvěřitelně - v současné době to je přes dva tisíce disket, které obsa- huji víc jak 8000 různých programů.

Takový vzrůst byl možný díky vyso- ké profesionalitě firmy při získávání a distribuci programů a v poslední do- bě také díky technickému pokroku v o- blasti záznamových médií.

V současné době se celý tento oh- romný balík programů nabízí a distri- buuje na jednom jediném CD ROM dis- ku za přijatelnou cenu. Edice se v pra- videlných intervalech doplňuje.

Adresa firmy je: PC-SIG, Inc., 1030 East Duane Avenue, Suite D, Sunnyva- le, CA 94086 USA.

V USA existuje i ohromné množství software pro operační systémy CP/M, MS-DOS, UNIX a Macintosh v několika archivech spravovaných složkou US Army Information Systems Command, White Sands Missile Range, New Mexi- co. Z Evropy lze tento software získat po připojení na mezinárodní komuni- kační síť InterNet nebo MILNET (pomo- cí FTP protokolu).

Firma FCC Folprecht má k disposi- ci jak PC-SIG LIBRARY, tak i PC-BLUE edici v posledních vydáních, tedy nej- čerstvější. Za pomoci sesterské spo- lečnosti v SRN má přístup do meziná- rodních počítačových sítí a díky tomu jsou pro ni přístupné ohromné ame-

rické programové archivy. Je tedy z čeho vybírat ...

Co vede firmu FCC Folprecht k aktivitě v oblasti Public Domain?

Není to každopádně činnost pro zisk! Ceny, za které se software rozšiřuje, pokrývají právě tak provozní náklady - tedy cenu diskety a plat pracovníků touto činností pověřených. V ceně již nejsou započítány další náklady, které firma musí vynaložit, aby se ke zdrojům Public Domain dostala (např. hodnota technického zařízení, komunikace po sítích, ceny optických disků atd.).

V Československu je ohromný nedostatek informací v oblasti výpočetní techniky, nedostatek odborné literatury i kvalitního a přitom levného software. Přitom je u nás dost chytrých a pracovitých lidí, kteří by mohli ve výpočetní technice hodně dokázat, ale právě tyto nedostatky jim „svazují ruce“.

Aktivitou v oblasti Public Domain chce FCC pomáhat tyto mezery vyplnit.

Vedlejším výsledkem jakéhokoli podnikání je reklama - ať chcete nebo nechcete, vždy si svojí činností nějakou reklamu děláte.

Aktivita firmy FCC Folprecht v oblasti PD jsou zatím následující:

1. Zasilatelská služba PD software

Public Domain Software, který si v katalogu vyberete, Vám firma na dobítku zašle na disketách.

Cokoliiv nového se ve světě objeví, zařazuje se do katalogů nejpozději do jednoho měsíce.

2. Vlastní edice FCC Public

Kromě toho, že Vám na žádost zašlou cokoliiv z programů Public Domain, připravují pro Vás pracovníci FCC vlastní ediční řadu disket s titulem FCC Public. Cílem této edice je pomoci Vám orientovat se v nepřehledné spoustě programů Public Domain. Vybírají za Vás ze všech dostupných zdrojů nejzajímavější a nejčerstvější programové novinky, o kterých jsou přesvědčeni, že by ve Vaší knihovně programů neměly chybět.

V edici se sídají především tyto tematické okruhy:

- programovací jazyky, uživatelské knihovny,
- databáze,
- CAD a kreslicí programy,
- programy pro vědecko-technické využití,
- systémové programy,
- BBS a komunikace,
- antivirové produkty,

- programy pro děti a mládež, vzdělávání,
- počítačové hry a zábava.

3. Přímá služba Public Domain

U firmy FCC Folprecht v Ústí nad Labem se buduje další pracoviště, na kterém si některé vybrané novinky, především z edice FCC Public, budete moci sami vyzkoušet a nahrát zdarma na vlastní přinesené diskety.

4. Hot line Public Domain

Hot line neboli horká linka je nová služba, která v ČSFR nemá dosud obdoby! Na zvláštním telefonním čísle bude připraven zkušený pracovník, který Vám ochotně poradí v oblasti Public Domain, poskytne zasvěcené rady, doporučí určitý program.

Nepokoušejte se ale nikdy pomocí této linky cokoliiv objednávat - objednávejte zásadně písemně nebo osobně u firmy. Hot line totiž bude sloužit především jako pomoc těm uživatelům, kteří mají nějaký problém.

5. Kupónová soutěž firmy FCC Folprecht

Ve spolupráci s časopisem Amatérské radio Vás chce firma pravidelně seznamovat s nejzajímavějšími novinami na trhu Public Domain Software. Součástí takové nabídky bude i kupón; zašlete-li spolu se žádostí o software i příslušný kupón, dostanete disketu se slevou, to znamená, že časopis, ze

kterého jste objednali, se Vám vlastně zaplatí. Navíc každý dvacátý žadatel novinky získá od firmy jednu disketu PD zdarma! Tyto výhody budou pro určitý software platit právě jen do doby než vyjde nová nabídka!

6. Public Domain československé výroby

FCC chce spolupracovat se čtenáři AR mnohem těsněji, především s těmi, kteří si myslí, že vytvořili nějaký zajímavý a užitečný program a jsou ochotni jej dát k dispozici všem ostatním. Proto se zakládá edice původního československého Public Domain Software a FCC vyzývá všechny ochotné autory ke spolupráci. Zašlete na ukázkou své práce, po posouzení budou nabídnuty naší a prostřednictvím partnerské organizace v SRN i zahraniční veřejnosti jako originální PD československé produkce. Je to jedna z možností jak ukázat světu, že i my máme chytré programátory.

* * *

Vážení čtenáři, aby firma FCC Folprecht dokázala na Vaše potřeby reagovat co nejpružněji a se znalostí oblastí Vašich zájmů, připravila pro Vás malý dotazník. Prosíme Vás - okopírujte a vyplňte jej a zašlete na uvedenou adresu firmy FCC Folprecht. Několik desítek došlých dotazníků bude slosováno a výherci dostanou disketu s překvapením! Výsledky této ankety poslouží i redakci Amatérského radia při výběru témat programů pro rubriku Počítačová elektronika, i pro výběr volně šířených programů, které budou v rubrice popisovány podrobněji..

Jak vás zajímají jednotlivé oblasti programového vybavení?

zajímá mne: moc málo ne			zajímá mne: moc málo ne		
textový editor	<input type="checkbox"/>	<input type="checkbox"/>	UNIX utility	<input type="checkbox"/>	<input type="checkbox"/>
tabulkový kalkulátor	<input type="checkbox"/>	<input type="checkbox"/>	programování assembler	<input type="checkbox"/>	<input type="checkbox"/>
databáze	<input type="checkbox"/>	<input type="checkbox"/>	programování jazyk C	<input type="checkbox"/>	<input type="checkbox"/>
věda, výukové programy	<input type="checkbox"/>	<input type="checkbox"/>	programování Pascal	<input type="checkbox"/>	<input type="checkbox"/>
matematika, fyzika, technika	<input type="checkbox"/>	<input type="checkbox"/>	programování BASIC	<input type="checkbox"/>	<input type="checkbox"/>
CAD systémy, desktop publishing	<input type="checkbox"/>	<input type="checkbox"/>	počítačové hry	<input type="checkbox"/>	<input type="checkbox"/>
kommunikace, sítě	<input type="checkbox"/>	<input type="checkbox"/>	domácnost, hobby	<input type="checkbox"/>	<input type="checkbox"/>
MS DOS utility	<input type="checkbox"/>	<input type="checkbox"/>			

Zájem, na který jsme zapomněli:

Jméno a příjmení:

Adresa (ulice, PSČ, místo):

Dosažené vzdělání:

pracoviště:

Vyplněný dotazník zašlete na adresu:

FCC Folprecht, spol. s r.o., Velká hradební 48, 400 00 Ústí nad Labem.

Obálku označte viditelně heslem FCC Public I

PUBLIC DOMAIN SOFTWARE

SEZNAM DOSTUPNÝCH PROGRAMŮ

Dostupný software je tématicky rozdělen. V levém sloupci jsou názvy programů nebo programových souborů. Čísla v pravém sloupci slouží při hledání žádaného produktu na CD ROM disku PC-SIG Library (April 1990). Programy si můžete objednat u firmy FCC Folprecht, Velká hradební 48, 400 01 Ústí n.L.

ACCOUNTING & FINANCIAL BUSINESS

Accounting, Billing

Fast Invoice Writer	1147
Fast Statement Writer	1216
Involcer, The	1348
ManageX I	0888
Micro Register	1806

Accounting, Checkbook Management - Business

PC-Check	0275
PC-Soft-E	1129

Accounting, Checkbook Management - Home

Bank Account Manager	1134, 1258
BANK STATEMENT TAMER	1435
BankBook	1911
Cash Control	1966
CCI Home Budget	1864
Charge Account Manager	1167
Checkbook Management	0393, 2049
Checkbook System	0397
CheckBooks & Budgets Plus	1126
Checkbooks and Budgets	0735
Checkease	1677
CheckMate	0784
Cheque-It-Out	1559, 1560
Chex	2015
CK System	0462
Exacct	1306
Exchequer	1786
ExpressCheck	1057, 2085
FamTRACK	0889
FastBucks	0855, 1296, 7063
Home Budget Management System	1302
Home Management II	1108
Home Money Manager Ila	0837
Mark's Checkbook Plus	1793
Multi-Assistant	1423
PC Account	0941
PC-Billmaster	1021
PC-Flow	0957
PC-Money	0532
Personal Ledger	1482
QCHECK3- Checking Account Management System	0968, 0969
QPacs	1640
Quick Check Book	1721, 1731
Time and Money	0251
DAC Easy Tutor	1294

Accounting, G/L, A/P, A/R, Payroll

Account + Plus	1871
Accounting 101	1193
Accounts Payable Lite	1545, 1546
Budgetrak	0519, 0520
C-A-S-E Accounting	1115

CheckMate-GL	0785
CPA-Ledger	0466, 0468
FCP Accounting	1985
Finance Manager II	0151
Finance Mgr II Accounts Receivables	1107
Freeway	0671-3
Medlin Accounting	0331
Micro Accounting System	2030
NonProfit General Ledger	2055
Painless Accounting	2059-61
Pay-Time Payroll System	0963, 1034
Payroll Lite	1547, 1548
PC-Books	0901
PC-Deal	1870
PC-General Ledger	0237
PC-Payroll	0565, 1019
Private Bookkeeper	
RAMbase Accounting	2033
Rosewood Journal	1309
SBW	1094, 1095
Simple Bookkeeper	0860
SOAR	0812, 0813, 1185
SPC-Payroll	1960
W211.BAS	1762
WYS-AR	1856

Accounting, Inventory Control

Easy Inventory	1996
Inventory Sort	0961
Inventory Master	2054
PC-Inventory+	1910
Pricebook, The	1373, 1374
SPC-Inventory Plus	1961
Starbooks	1250
Stock Inventory Control	0958

Accounting, Job Costing and Bill of Materials

Cost Calculation	2035
Cost Effective Tracking System	0845
COST-BIZ	1984
Job Cost	1230
Takoff	1715
Purchase Order System	0741

Accounting, Time-Billing Manage- ment

Billing Manager	0909, 0910
BillPower Plus	1168
Mr. Bill	0469, 0470
Time Tracker	0825, 1024

Business Specific Systems

Advertising Response Manager ARM	1562
Call Master	1622, 1623
Chiro Patient Tracking System	1044
Construction Estimator	1159
Dental P.M.S	1251-55
GIFTBASE	1991, 1992
PC-FOOD	1809
ReSearch (formerly Litigex)	1272

Service Plus	1270, 1271
Winning Bid, The	1388

Form Creation Programs

EZ-Forms Executive	0404
EZ-FORMS FIRST	1838
EZ-FORMS-LITE	1202
Form Master	1099
Formgen	1561

Investment Management

APEX Financial Markets	2038, 2039
APEX Futures & Commodities	2040, 2041
APEX Mutual Funds	2042, 2043
APEX Point & Figure	2046, 2047
APEX Stock Options	2045, 2044
Finance + Plus	1873
Finance Plus	1919
MarketPLUS	1153, 2005
Owl Personal Portfolio Manager	1639
PC-CHART	1436
PC-Chart (EGA Version)	1704
PC-Portfolio	1310
PC-Stock	0575
Personal Portfolio Analyzer	2006, 2007
PFROI	0360
PORTRAC	0754
Portwort Package	0101
Security Analysis & Portfolio Manager	1684
Stock	1159
Stock Charting System	0246, 1354
Stock Charts	1528
Stock Market Timer	1800
The Manager's Toolbox	1928
TRANSTOK	0323

Loan Calculators

AMORT	1646
Amort70	1805
Amortization Table	1027
Amortization Calculator	1693
AZ Real Estate Finance	1882
Banker	1159
Dollar-Pro	1012
Finance Analyzer	2016
Financial Calculator	0994
Genamort	1874
INTCAL	1310
Loan Warrior	1055
LOANCALC	0960
LOANPMT	1854
LoanStar	1580
Monamort	0960
MortPlan	1342, 7084
PCLOAN 4	0399
Pelton Computer Consultants	1944
Varamort	1323
Your Financial Analyst	1613

Real Estate and Property Management

Crisp	1346
Mini-Minder	0961
OverLord	2008, 2009, 2010
PC-Agent	1405
Professional Real Estate Analyst	0923
PropMan	1179, 1180, 1452
Real Estate Systems	0656
RealQuik	1620
REIPS	0860

Quotes, Bids, and Estimating

BID-BIZ	1855
Easy Quote	1812
PC QUOTE	1575
PC-Bid+	1955
PC-Estimator	1383, 1384
PDS*Quote	0533
SC-Estimate	1715

Tax Preparation

AM-Tax	0479
PC-Payroll W-4 ModuleTax-Planner	0982

BBS & COMMUNICATIONS

Bulletin Board Systems (BBS)

Electronic information System	1627
FindBBS	1370
Fora	0948
Mubbs	1458
RBBS-PC	0212, 0334, 0621, 0622, 2092
Sapphire	1833
Search Light BBS	1136, 1137
Var-Tale	1794
Wildcat BBS	0745, 0746

Communication Programs

BACKMAIL	1841
Boyan Communications	1206, 1343
Brain, The	0752
Brainstorm!	1927
BSR	1987
Dialer	0717
FoneBook	1226
GT Power	0782, 0783, 1549, 1550
Ideal Terminal (IT)	0679
KERMIT	0041, 0042
Message Master	0988
Omega-Link	0843
One-To-One	1031
Private Line, The	0893
Procomm	0499
ProFonEdit Plus	
- Procomm Plus Directory Editor	0956
QMODEM SST	0310, 1022,
	1023, 1483, 1484, 7076
QuickTran	1371
Serial File Copy	2003
Supercom	0717
TapCis	2051
Weak Link	0893
XPort	1775

DATABASES & MAIL LISTS

Database, dBase Utilities

dbScan	2098
DBSCREEN	0163
dBSearch	1995
dEdit	1569
dLITE	1204
dProg	2036

EZWin	1947
Intelli-Trieve	0811
XTAB	1937

Databases

A2Z	1740, 1741
ADDTel	1674
Classic Solutions	1138, 1139
Database Publisher	0854
FFD	2058
File Express	0287, 0288, 7064
Find-X	1719
Free File	0521
I Found It	1607
Itemized Calculator, The	1375
Multi Base	0971
Multifile	0835
Muses	0635, 0636
NewBase II	0233
Newsbase	0715
PC Data Controller	1723
PC-File	2082, 2083, 2084,
	7411, 7412, 7049
PDS*BASE Database System	0396, 1025
Pirouette	1746, 1747
Profile	1002, 1003
Quiksort	1152
Simplicity	1581
Snap Filer	1043
Wampum	0830
WYNDFIELDS	2120
ZoomRacks	1287, 1288
ZoomRacks Business Kit	1289
ZoomRacks Home Starter Kit	1290
ZoomRacks Small Business Kit	1291

Mail Lists, Address Managers, and Telephone Organizers

Address Book	0988
Address Manager	0218
Bulkmail	1527
CataList	0864, 0865
Directory Assistant	0748
Dmail	1172
Doctor Data Label	0943
Easy-Plan	1541
Fone: Business	1729
Fone: International	1728
Fone: Residential	1730
Letters n Labels	1823
Lst-Mgr	0935
Mail It!	1064
Mail Monster	0483
Mailing List Management DataBase	1702-3
Mailing-Made-Ez	1795
MailList	1506
Mass Appeal	1920
MAST Mail	1564
MEMBERSHIP LIST	1696
NamePal	0706
PC-Mail	0868, 0869, 1475, 7072
PC-Postcard Plus	1361
PC-Speedy Mailer	1557
Phone	1726
Phone Caddy	1154
Pocket Filer & Pocket Notes	0896
PRO-DEX Database Management	0927
Reliance Mailing List	0503
Zip Appeal	1921-3
Zipkey	1767

Maintenance, Recordkeeping

EIMS	1843
Maintenance DBS	1625

ENTERTAINMENT & GAMES

Entertainment, General

Crossword Creator	0819
Do-It-Yourself Promo Kit	0996
Fictionary	2087
G.I.F.T.S.	0876
Garden City Software Collection #1	1421
Garden City Software Collection #2	1422
Get Lucky!	0711
Greetingware for	
Christmas and Birthdays	0839
It's All in The Baby's Name	1214
MOSAIX	1972, 1973
Origami	0408
Thinkle	1691
Tommy's Insults	1736
Universe Analyzer	1772

Gambling

Cambridge Thoroughbred Handcapper	0918
Casino Slot Machines	1573
College Football Pool	0892
Football Fun	0747
Football Pool Manager	0882
Horses	1514
Klondike	1544
NCAAPool	1387
NFLPool	1387
Slot	1544
SportsBook	1132
Track Man Database Manager	1660

Games, Adventure and Text

Adventure Game Toolkit	1231, 1232
Adventure Addiction (unsupported)	
Adventureware	0453
Alice in Wonderland	1495
Basketball Simulation	0946
Battle Ground	1090
Beyond the Titanic	0832
Big Three	1925, 1940
Cavequest	0451
Dark Continent	0967
DNDBBS	1131, 1676
DRACULA in London	1220
DUNGEON	1437
Dungeons of the	
Necromancer's Domain	0567
Eamon	0296, 0297, 1038, 1039
Facing the Empire	1075
FailThru	1389
Finnish Games	1536
Frigate	1467
FRP Game Master Utility	1063
GameScape, Giant Space Slug,	
Marooned & Ringwielder	1293
Golden Wombat of Destiny	0678
Hack Keys	1417
Hack, The Amulet of Yendor	0452
Hogbear Adventure Game	1269
It's A Crime!	1825
Kingdom of Kroz	0832
Landing Party	0604
McMurphy's Mansion	1073
Megagopoly	1194
Mix It Up	0911
MORAFF'S REVENGE	1641
Nebula	0967
NETHACK: An Adventure Game	1000
Nirvana	1281
Prince	1140
Quantoids of Nebulous IV	0749
Red Planet	1073
Secret Quest 2010	0633
Sleuth	0694

Super Nova	0952
The Asian Challenge	1688
THE SOCCER GAME	1695
Time Traveler	1075
Under the Gulf	1852
World Generator	1133

Games, Arcade

Adventures of Captain Comic	1450
Allen Worlds	1939
Arcade Series 3	0293
Asteroid Field Battle	1523
BEGIN A Tactical Starship Simulation	1692
Brian's Games	0274
EGA Trek	1221
Foilies	0514
Game Series 18	0390
Game Series 21	0456
Games Galore	1328
Greatest Arcade Games	0457
International Game Collection	1140
LINEWARS	1764
Maze Cube	1419
Miscellaneous Games	0945
Patrick's Best Games	0476
Paul's Games	1814
Pinball Rally	0557
Racei	1892
Return to Kroz	2001
Space Rescue	0749
Super Pinball	0723
Tommy's Gorilla Balls	
/Tommy's Meteors	1604
Tommy's Saucer	1282
Tommy's Trek	1718
Willy the Worm	0445
Worthy Opponent	1175
Wreck Tangle	1836

Games, Arcade (Sports)

Basstour	1319
Budget Baseball	1400
Cunning Football	1240
FOOTBALL	1990
PC Pro-Golf	1344
ProChallenge Baseball	1207
Statistically Accurate Baseball	2074

Games, Board and Dice

AGGRAVATION	1590
Backgammon	0708
Bingo	1913
Board Games	0993
Crazy Shuffle	2014
Five Dice Game	0743
Kakal	2013
Liberty Bell Slot and Wheel	2064
Mahjong	0641
MahJongg	1221
Miscellaneous Games and Lotto	1292
NCRisk	1322
OTH	1523
PC Bingo	1897
PC-Areana	1518
PC-CHESS	0120
PC-Jigsaw Puzzle	1558
Pig	1733
SPRINGER	1434
Triple Othello	1556

Games, Cards

20th Century FARO	2088
BlackJack	1329
BlackJacki	1283
Bridge Game, The	0890
BridgePal	0780
Card Game Collection	1280
Computer Contract Bridge	0713
Computer Draw Poker	0714

Computer Solitaire	1177
Cribbage By T	1789
CRIBBAGE PARTNER	1819
Draw Poker Professional	1962
Game Fun for All	1074
GAMES FOR EGA	1938
Golf Solitaire	1700
Lowball Poker Professional	1956
Osmosis Solitaire	1913
Poker	0791
Poker Challenge	1642
PSQUARES	1191
Pyramid Solitaire	1701
SMSPOKER	0055
Solitaire	0891
Solitaire Card Games	1411
Tommy's Gin Rummy/Meteors	1612
Ultima21	0791
UNO and Games	1644
Vegas Johnny's Draw Poker	1896
Vegas Pro	1490

Lottery

Lottery Analyzer and Picker	1750
Lottery Fun	0749
Lottery Player iii	1485
Lotto BlockBuster	1735
Lotto Challenger	1906
Lotto Fever	0929
LOTTO MAGIC WHEEL	1552
Lotto Master Professional	1318
Lotto Prophet	1006
Lotto Prophecy	1616
Lotto-Magic	1815
Lotto-Trax	1391
LOTTOplks	0789
Smart Money	1329
Super Lotto-Master	1407
The Winning Edge-Lotto	1865

Music

Christmas Concert Volume 1	1211
Christmas Concert Volume 2	1212
Composer	0722
Composer	0794
Lap Dulcimer Tuner	
and Sheet Music Sampler	0967
Music Collection	0322
Music Minder, The	1472
PC-MUSICIAN	0127
PIANO	0322
Planoman & Player Piano	0279
Planoman Does Beethoven	1396
Planoman Goes Bach	1395
Planoman Goes Baroque	1397
Speaker	2019

PCjr

Amy's First Primer Jr	0647
PC-Calc Jr	0625
PCjr Educational Games	0241
PCjr Games	0354
PCjr Survival Kit	0999

Photography

Computer Darkroom	1313
For Photographers	1164
PC-Foto	1040
Photo Pack	1249
Reorders and WedPrice	0937
Slide Manager	1390
Slide.PC	1418

Practical Jokes

A_Curse	1523
CrazyDos	1912
Hotboot and Insults	0619
LoveDOS	1403

Program Grab Bag	1314
TV/VOLAPUK	1912

Spelling, Word, and Vocabulary Quizzes and Games

ARIONX: Seeker of Gans	1090
Flash Cards: Vocabulary & Spelling	0367-70
Foreign Language Hangman	0708
Gramarcy	1158
Guess IT II	1563
Hangman	0153
Hangman by Victor	1965
Letter Shift	1556
NameGram	0477
PC CRYPTOGRAMS	1964
PenName	1710
Phrase Craze	0385
Spell Games	1144
Tommy's Crosswords	1711
Tommy's Wheel of Misfortune	0891
VMS Vocabulary Management Sys.	1379
Word-Part Dictionary	1534
Words	1998, 1999
Words*Words	1897

Sports Management

BlkelInfo	2017
Bowl - 101e	1246
Bowling League Secretary	0023
Bowling League Secretary's System	0884
BowlSTAT	1953, 1954
Diver's Logbook	2050
Double Bogle Killer	0984
Fantasy Sport Drafter	2078
Fish-N-Log	1659
FishBase II	1358
Foursome Generator	1174
Golf Handicapper	1827
Golf Scorekeeper	2086
Graphic Coach	0849
High Points	1315
PC-Golf	0262
PC-Sport	2079
PRO FOOTBALL LINEMAKER	1905
SailScor	1801
SimBase	1186, 1187
Sports League Management	1844, 1845
Sports Statbook, The	1420, 1427

Trivia

EZK-Quizzer	1449
Opus 1 Brain Teasers Trivia Game	0997
Trek Trivia	1278
Trivia	0327, 0328
Trivia	2000
Trivial Towers	0473, 0474
Tune Trivia	0475

GRAPHIC, DRAWING, CAD

CAD (Computer Aided Design) and Designing

Curve Digitizer EGA/CGA	2066, 2067
Curve Digitizer (Hercules)	2068, 2069
DANCAD3D	0701-04
Draft Choice	1760
Droege	0904, 0905
EDRAW	0828
PC-Draft II	1060
PC-Draft-CAD	1757
Quest 3-D	0762
VGACAD	1587, 1588

Clipart

Cooper Graphics #01	2100
Cooper Graphics #02	2101
Cooper Graphics #03	2102

Cooper Graphics #04	2103
Cooper Graphics #05	2104
Cooper Graphics #06	2105
Cooper Graphics #07	2106
Cooper Graphics #08	2107
Cooper Graphics #09	2108
Cooper Graphics #10	2109
Cooper Graphics #11	2110
Cooper Graphics #12	2111
Cooper Graphics #13	2112
Cooper Graphics #14	2113
Cooper Graphics #15	2114
Cooper Graphics #16	2115
Cooper Graphics #17	2116
Cooper Graphics #18	2117

Graphing Programs

ExpressGraph	1058
GraphTime II	0669, 0670
GraphTime-II	0833, 0834
Screen-Print Bar/Line Graph	
ZGraf	1967

Paint and Drawing Programs

Charts Unlimited	1496, 7085
Drawplus	0633
Fantasy	1305
Finger Painting Program	0763
FloDraw	0912, 0913, 2020
GRAPH	1909
MacPaste	1001
MonoDraw	1092
Painter's Apprentice	1645
PARTYDOT	1553
PC-Art	0629
PC-Key-Draw	0344, 0345, 1032, 1124
PictureThis disk 1/2	1130, 1474
Printer Art	0154
Ready Set Draw	1851
Spilo	1986
Sprogh	1524
TurboFlow	1824

Presentation Graphics (Slides)

Collage	0975
CompuShow	1461
Gemcap	1305
PC-Demo Graphics	
Presentation Package	0914
PC-FOIL	0347
Slide Generation	0244

HOME & PERSONAL

Auto/vehicle Management

PC Mechanic	1277
Schedule Magic	1098
TLC Truck Data System	1085
Vehicle Expenses/Maintenance	1748

Food and Drink Preparation

Computer Baker	1171
Edna's Cook Book	1229
Grocery	0856
Home-Bartenders Guide	1516, 1517
PC-SIG's CookBook Volume I	1215
Recipe Index System	1317
Recipes	1210

Health Management

Biorhythm Monthly Schedule Prog.	0879
Diet Disk and Weight Control	1077
DIET-AID	1578, 1579
EKG/Tracings/Pump	1513
Health Risk	0192
Insulin	1311

Kinetics	0947
Managing Your Food	1056
MEALMATE	0700
Nutritionist, The	1148, 1675
Parents Home Companion:	
Managing Collc	1100
PC-NurseWorks	1776
Pregnant	1325
Slimmer	1489
Weight Control	1325

Home Management

BTU Analysis	1832
Coupon	0724
Garden Productivity Calculator	1894
Gardener's Assistant	1125
Home Applications	0321
Home Insurance	0796
Home Inventory Program	1349
Home Inventory Record Keeper	1811
Home Maintenance and Repair	2072, 2073
Linear Optimizer 2	1062
Point & Shoot Home Manager	0940
QuickList	1033
Refund	1323
Rokmar HouseHold Inventory	1227
Room	1898
RootDirectory	1941, 1942, 1943
Shop	1648
The Christmas Helper	1840
The Weather Channel	1903
Wine Cellar	2080
YOUR VERY OWN Home Inventory	0395

Movie/VCR/Music Databases

Abstract	1300
AUDIO II	1438
Becknervision Video	
Database Manager	1030
CassetteI	1734
CDMASTER	1439
db-VCR	1347
Disk Drive-In Theater	0493
For Record Collectors	1104
Home Movie Librarian	1266
Movie Database, The	0311, 0774
Record Finder, The	1276
Records & Tapes Database	1413
ShowLog	1274
Video Librarian	1424

CHILDREN & EDUCATION

Children's Programs (ages 2-10)

ABC Fun Keys	0844
Amy's First Primer	0646
Balloon Speller	1333
Boat Box	1466
Brandon's Lunchbox	1465
Fact Pack	1381
FlashBox	1837
Fun With Letters and Words	1278
Funnels and Buckets	0229
Googol Math Games	1768
IQ Builder	0018
Jem	0922
Kid Paint	2065
KidGames	0705
KinderMath	1629
Learn to Guess	1412
Math Castle	1908
Math Tutor	0095
Number Magic	1214
Play 'n' Learn	0916
USA Box	1857
Wizquiz	0922
Word Processing for Kids	0343

History, Education

Presidents, The	1065, 1066
-----------------	------------

Language Study, Foreign

Cantonese Tutor	0755
French Verb Conjugator	0758
Hebrew Quiz and Tutorial	0902
Japanese Tutor	0712
Language Teacher	0612
Spanish For Travelers	1061
Spanish Verb Conjugator	0757

PRODUCTIVITY

Desktop Organizers

Alt	1189
Compass Desktop Manager	0764
Ez-Desk	1256
Hal9000	1242
IMX Co-Pilot	1020
PC-DeskTeam	0405, 7040
Power Desk	1738

Project Management

Automated Planning Form (APF)	0955
BestGues	1889
Easy Project	1082
Gantt Charting	1306
Gantt Pac	0593, 1198
PC-Project	1340, 1341
Production Control Schedule	1244
Project Management	0423
RKS Task Master	1380

Personnel Management

Employee Management System	0660
Hire	1004
Management Mentor	1156
Managing People	1316
PC Shift	1195
PC-PUNCH	1589
The Complete Office	1585, 1586

Resumes and Job Search

Apply	1005
Resume Shop	1667
Resumebest	1097

Sales and Prospect Management

Contact Tracker	1880
Front Office, The	0697, 0698
IN-CONTROL	0687, 0688
Phoebe	1790
Sales Call Reports	1610
Sales Tools	1582
SALES-BIZ	1839

Schedulers, Calendars, To Do Lists, and Ticklers

Active Life for MS-Windows	1481
Active Life for MS-DOS	2022, 2021
Active Life for OS/2 PM	2023
Ample Notice	0872
APPTTracker	1792
Blg Event	1900
Calendar Mate	1907
Calendar Program	1571
Calendar Program by Small	1963
DFStickI	0818
Flexical	1106
H&P Calendar	0995
Instacalendar	0709
Judy, the Memory Resident Calendar	0848
LCQ/Remind	1867

MakeMyDay	0618
Moe	1225
MOMSYS	1716
Names & Dates	1566, 1568
PAL	1259
PHONEM12/COMTRC11	1530
ProDex Plus	1248
Remember-It	1217
Reminders	1162
Supercal and EGAcad	1308
Thl User Library	1114
TickleX	0934
Tikler	1571
ToDo	0917
TrackStar I	1963
Year Planner	1626

PROGRAMMING, AI & COMPUTER EDUCATION

Artificial Intelligence

Decision Analysis System	0953
ESIE	0398
EyeSight	1678
Imp Shell, The	0761
Management Action Expert	0976
XLISP	0148
XXXpert	0883

DOS AND PC TRAINING and Education

DOS Practice	1950
DOS-Pro	0889
DOSHelp	1067
Help/Pop-Help	1881
HelpDOS	0686
PC-DOS HELP	0254
PC-Learn	1399
PC-Prompt	0558
PCHELP & Utilities	1774, 2026
QRef	0875
Turbo-Lessons	1932
TUTOR.COM	0403

Hypertext

Black Magic	1120, 1121, 1122
EVERGLADE	1901
Hyper Helper	1332
HyperShell	1720
Hypersketch	1473
Hytext	1234

Programming, Assembly

A86/D86 Assembler/Debugger	1111
Assembly Utilities	0309
CHASM	0010
Doug's Programming Language	1262
Hextodec & Dectohex	1650
PC/370 Cross Assembler	0402, 0859, 1352
PseudoSam 18 and 65	0776
PseudoSam 48 and 51	0777
PseudoSam 80z and 85	0778
PseudoSam 68 and 685	0775
Sofa	2058
TASM	0643

Programming, BASIC

ACSORT	1959
BASIC Games	0174
BASIC Games & Programming Intro	0595
BASIC Language Games	0045
BASIC Programs	
In Finance and Inventory	0171
BASIC Programs	
In Math and Statistics	0180
Basic Windowing Toolbox For BASIC	0527

BASICXREF	0358
Baslin	0989
BetaTool's BASIC Development Syst.	0269
CrossRef	1968
Designer, The	0069
DISAM	1617
EDFIX2	1959
Finance Calculator	0029
Funky ToolBox	0907
Managing Your Money	0613
PC-Professor	0105
PC-Talk III	0016
QB4CREF	1959
QBSCR Screen Routines	1714, 1713, 1712
Reformat	1842
Sounds In BASIC	0053
Structured Programming Language	0666
VRef	0875

Programming, C

C Adventure	0259
C Tutor	0577, 0578
C Utilities, Volume 2	0216
C-Window Toolkit	1007, 1705
Elementary C	0429
Extended Library	1324
FlashPac C Library	1872
Indent	0962
MPLUS Graphic Interface Library	2032
Panels For C	1717
PC GRAPHICS C	1433
PC-GRAPHICS C	1433
PCC Personal C Compiler	1337
SDB	0147
Steve's Library for Turbo-C	1737
Turbo C Tutorial	0816, 0817
Turbo Enhancement Toolkit for C	1268
Utilities for Programmers	0885
Window Boss	0873, 1113

Programming, Forth

D-Access	1930
Graph By Hoyer	1926
Laxon & Perry Forth	0263, 0264
MVP-FORTH	0031

Programming, Fortran

ACM Volumes 1-5	1008, 1009, 1010, 1011
Automatic Fortran Format Statements	1685
GRAFX - Graphic Extension Library for FORTRAN 77	1699

Programming, Modula-2

Calutil	1649
M2JFTOOLS	1895
Modula-2 Compiler	1080, 1081
Modula-2 Tutorial	0814, 0815

Programming, Pascal

DML & XREF	1697
Eislinger Programmer's Utilities	1539
FlashPac Pascal Library	1982, 2093
Minigen	0760
MultiTrack	1744, 1745
Mystic Pascal	0965
P-Pascal Translator	1321
P-Robots	1386
Pascal Turbo Enhancement Toolkit	1247
Pascal Tutor	0579, 0580
PXL	1304
ScrDesign	1092
Student Pascal	1321
T-Ref	0827
TechnoJock's Turbo Toolkit	1651, 1652
Turbo Apprentice	1459
Turbo Designer	1353
Turbo Pascal Modulating Printing Filter	1531

TurboMenu System	1983
------------------	------

Programming, Tools

ADA-TUTR	1529
BOX	0842
CAPP	1176
Colorset	1370
ColorSet	1808
Creating User-Friendly Software	1609
Extended Batch Language	0124
Fast	2058
Fixer	0978
Font Editor	0856
Fonted	2096, 2097
Hacker	1902
HexEdit	2019
Icon Maker	0485
Loadkey	1834
MMake	0978
Model-S	2070, 2071
OLV312	1933
PC-Tags	2011
Personal Apt	1722
PopText	1935
Pro-Inst	1765
Probat I	1565
Qparser Plus	0419
SCRDES	1303
Screen-Do	1118
ScreenPaint	1510
SEBFU	2081
Snobolyt Utilities	0928, 1128
SPA:WN	0442
Temescal	1952
TesSeRact	1491
Turbo ScrEdit	1653
UnlScreen	2056, 2057
Vanilla Snobol4	0980
X-Batch	1619

RELIGIOUS & REFERENCE

Bible Study

Bible Men	0781, 7082
Bible Quiz Plus	0974
Bible, The	0766-72
Bible-Q	0628
Daily Bread	1327
Destiny	1327
Gospel Concordance	1208
King James Bible Search Program	1654-57
Problems	1526
Revelation Tutorial	1554, 1555
Scripture Memory	1658
SCRIPTURE	1816, 1817
SeedMaster	1591-1600
Verse	1511
WordWorker	0581, 0582

Church Administration

Church Accounting System	1798
Church Membership System	0742
Church Shareware Software	1753
Church Treasurer	1538
Heritage Church System	0881
Minister Database	0954
Minister's Sermon Indexer	0790
Red Squirrel's Church Accounting System	1583

Library Record Keeping (Including Dewey Decimal)

Book Minder	1178
Cassy	1091
Pro Librarian	1313
Textbook Inventory	1532

Reference Materials (books/articles/information/data)

Baby April	1488
Ballistic	1802
Business and Moral Values, 4th Edition	0506, 1018
Code-Blue	1083
Damn the Trivia	
- Our Priorities are Drowning	0972, 0973
DIVORCE - Animated Strategy for Men	0886
Economic Research's Cost of Living	1401
Education of Handicapped Act	1690
ERB	1315
EthInves	1307
FBLCC - Building Life Cycle Cost program	0572
Filler Finder	1047, 1048
Findata Corporate Profile	1051, 1052
Future Systems	1160, 1161
Govern-US	1666
IBR HEATLOSS	1647
Maptab World Culture Database	0920
NBSLCC - National Bureau of Standards Life-Cycle Cost	0983
Radiation	1432
Radio Repair	1432
ReadFast!	1630
Select-A-College	1345
ShareDebate International	2121
Storm	1372
Terra*Time	1127
The Electric Almanac	1724
The Legal Pad	1624
Virtual Library, The	0979
Wisdom of the Ages	1498-1501
World Time	1669
Zip Insulation	1468

SPECIAL APPLICATIONS

Agriculture

Agriculture Assortment	0459
Agriculture Programs Ridgetown Coll	0461

Astrology and Fortune Telling

Astro195	0966
Crystal Ball	1492
Geomancy	1275
Mayan Calendar	1520
Procon Fortune	1054
Rune-Caster	1173
Sage Past Life Tutor	1486
Soothsayer	0615

Conversion Programs

Conversion Program	0989
CONVERT	0960
Convert	1183
Units	1014

Flying and Navigation

CAT	1681, 1680
Flight	0939, 1476
Freequent Flyer	1365-7
Passing Passages and GS-Testit	1362

Genealogy

Brother's Keeper Genealogy	1504, 1505
EZ-Tree	1611
FAMILY HISTORY	0361, 0632
Family Ties	0465
Family Tree Genealogy	0240
Family Tree Journal	1535, 2094
Genealogy ON DISPLAY	0090
Micro-Gene	1068

MyFamily Notes and Sources ON DISPLAY

1936
0594

Ham Radio

Conlog	1885
Ham Radio	0436, 0437
Ham Radio	1420
KBOZP MESSAGE HANDLER	1887
KBOZP Super Contest Log	1096
MiniMuf	1315
Morse	0939
Morseman	1759
PC-Ham	0562
SamfB - The Morse Code Program	1886

Maps

The Loran Locator	1542
World City Distance Computer	1127
World Digitized, The	0494-6
World29	1392

Survey Tracking

Survey System	1035, 1036
---------------	------------

Visually Impaired, Programs

Catcher	0732
Enable Reader Professional Speech System	0674-77
Impaired Laser Font	1223
Tracker for the Visually Impaired	0732

SCIENCE / MATHEMATICS & TEACHING

Astronomy and Space Exploration

Apollo Mission	0987
Astronomy Star Catalogue	0851, 0852
Astronomy Programs	0921
Astrosoft Ephemeris (ACE)	0692, 0693
Deep Space	0866, 0867, 2063
Exploring The Cosmos	1621
Launcher	0945
Moonbeam	0538
Optics	0538
Particle Simulation	1070
Planets	0298
Rocket Programs	0921
Satellite Programs	0921
Silicon Sky	1103
Skyclock	1614
Space Shuttle Tracking System	1689
The Night Sky	1796, 1797

Chemistry, Biology, and Physics

BSIM	1725
Chemical	0938
Crystal	1469
Electron	1045
Labcoat	0932
MSFORMS	0970, 1453
Nuclear Magnetic Resonance	0590
PC-Calib	1078
SCI-Calc	1326

Engineering

A-Filter	1799
CC-Surveyor	0926, 1029
Chart	1761
CoGo	1013
ResisPop	1884
Survey Land Yourself	1826
The Electrical DBS disk	1977-81

Fractals And Mathematic based Graphics

Cell Systems	1109
Life Forms	0802
Mandelbrot & Julia Set Generator	1076, 1241
Mandelbrot Magic	0841
Mandelbrot ZoomLens	1326

Math and Geometry Programs

Algebrax	1456
AnyAngle	1756
Are You Ready For Calculus?	0858
Curvefit	0707
DataPlot	1059
Differential Equations and Calculus	1072
EPISTAT	0088
Equator	0249
Evaluate	1904
Interactive Matrix Calculator	0823
Laser Graph	2027
LSTSQR	0925
MATH PAK III	0394
MATHPLOT	1863
Mtool	1183
Plot	0978
Plot	1013
Polymath	0606
SPlot	2024
Vibrating, Rotating, and Cooling	1182
Vision Free Software	1378
XYPlot	1515
XYSee	1507, 1508
XYsolve	1493

Statistical Programs

Kwikstat	0654, 0655
Micro Statistics Package	1931
Probability and Statistics	0985
STAT	1618
Statistical Consultant	0949
Statmate/Plus	0861-3
Stat	0990-92

Teaching, Grading Systems and Record Keeping

Aelius Gradebook	1239
Class Record	1071
Classbook Deluxe	0951
Gradease	1862
Grader	1071
Grades	2025
Gradescan	1233
Grading Assistant	1632, 1672
Mark Record Plus	1512
Teacher Works	1521
Teacher's Database	0878
The ProGrade System	1778, 1779
The Progress Report	1633
VAR Grade	0903, 1357, 2099

Test and Quiz Preparation

Exam Bank	1631
PC-CAI	1470
PC-Quizzer	1143
Pop Train	1471
Test and Train	0797
Test Management System	1890, 1891
TMaker	1551
Total Recall	1917

SPREADSHEETS & TEMPLATES

Spreadsheet Templates

FITT Lotus 1-2-3 Tax Worksheets	0290
---------------------------------	------

Lotus Financial Worksheets	0304
Lotus Worksheets and Macros	0301
Scientific & Eng.Tools for Lotus	1988
The Retirement Financial Projector	1875
Whiterock Alternative	0583

Spreadsheet Utilities and Education

Lotus Learning System	0846, 0847
Menu-Range for Lotus 1-2-3	1989
Pivotl	1763

Spreadsheets

AsEasyAs	0751, 7069
ExpressCalc	0524, 0525
EZ-Spreadsheet	0695
FreeCalc	0574
InstaCalc	0710
PC-Calc+	0199, 1016, 1017, 7036
Power Sheets	1284, 1285
QubeCalc	0696

UTILITIES - DOS, PRINTERS MISC.

Archive/Compression Utilities

Arc	0609
Archive Menu	1205
Compress	1037
DStuff	1861
LHarc & Utilities	2018
PKPAK, PKUNPAK, and PKSFx	1330
PKZIP, PKUNZIP, PKSFx	1364

Banner Makers

Banner	0386
PC-Banner	0779

Barcodes

BarCode	0877
---------	------

Copy Utilities

Dcopy	0936
Disk Duplicator	1451
DiskDupe	1279
EZ-COPY-LITE	1201
Formgen's Disk Duplicator	1694

Disk Catalogers and Disk Labelers

BJ-FILES	1804
Catalog	1312
DBS-KAT File Librarian Package	0537
Disk Cataloging Program	1049
Disk File	1356
DiskCat	1320
Diskette Manager	1393
Diskmaster	0709
DiskScan	1286
Disk Tag	1226
FBNCAT	1934
FlopCat	0959
Masterfile Disk Management System	0981
PC-Disk Label	1192
PmCat	1997

DOS Shells

DFM	1899
Direct Maint	1497
Directory eXtended	1605
Disk Navigator	0810, 7059
DiskWiz	1257
DOS Controller	0944
DOSamatic	0498
Edlcom-12	1037
EzDoss	1803
File Access	1860

File Friend	0871
File-Man	1199
FileViewer	1503
FLIST/PC	0933
Hard Disk Director	1219
IDCshell/NARC	1205
Job Organization and Backup Syst.	1093
P-Cam	1464
Point & Shoot	0930
Power Shell	1878, 1879
Quick Directory	2081
Roam	1151
Scout	0895
Scout-EM	1706
ShortCut	0840
Still River Shell	0481
StupenDOS	1336
Task Master	1635
Treeview	1243
VDOS	1914
Wyndshell	1190

File Management Utilities

Analyze	1619
ArgaUtilities: Argafind,-Menu,-copy	1359
Baker's Dozen	0800
Best Utilities	0273
Circle Software Utilities	1331
CopyAll	1503
Dabutil	1176
Direxit	0908
DIRUTILS	1853
Disk Utilities	1914
DOS Extensions	0585, 0586
DOS Utilities 16	0319
Dos-Ez	1994
Eddy	1026
EDITRAND	1918
EZDO	1176
FC and FCDOC	1584
File Patch	1945
Find Duplicates	2029
FindFile	1425
Findzz	2052
FreePack	1727
Fugue Utilities	1634
Fugue Utilities #2	1755
Garth's Shareware Utilities	1859
GetFile	1503
Grep	1929
Handy Dandy Utilities	1834
Hyde	0877
KYM-Disk Utility	1904
Leoce Toolkit	1533
LightWave Utility Disk	0736
ListBack and LstBack2	1994
Locate	0731
Management Utilities	1608
MASDIR	1876
Master Key	0598
Match-Maker	1791
Miscellaneous Utilities	0906
MultiBak	2052
PC-Util	1751
Process	1305
Read My Diskl	1394
Read.Com	1687
SetAttr	1503
Sorted Directory	0933
SUP	1406
TwoDisks	1636
Utility Potluck	1101
What	1305
WiseDir	1503
Xanadu Utilities	0737
XCUTE	0818
Xdir	0724
XDOS	0444

Hard Drive Utilities

BAKtrack	1869
Hard Disk Utilities	0478
Hard Disk Management	0786
Packdisk Utilities	0610
Point & Shoot Backup/Restore	1188, 7075

Label Makers

Easy Labels	1683
Fancy Label Maker	0871
Label CMT	1297
LABEL MAKER	0284
Label Master	0750
Label Utilities	1866
LaserLabel	1769
LaserLbi	2028
MR. Label	1673
PostNet	1758
PRTLLabel	0744
Simply Labels	1150, 1679, 7083

Menu Programs

Advanced Menu	1050
Automenu	0608
Easy Access	1409
EasyDOS	1176
EasyMenu	0738-40
EZ-Menu	0444
Front End	1301
Hard Disk Menu IV	0631
Instant Access Menu System	1577
Magic Menus	1196
Master Menu	1969
Menu Construction Set	1924, 2119
Menu Master	1043
Menu Master By Morgensen	1948
MENU, THE	0960
Menu-Matic	1141
MenuEase	1835
Menueze	0936
MenuShow	1410
MenuX	1788
Mr. Menu	0944
Mymenu	0936
Navigator, The	1042
Navigator, The	1279
Nifty	1709
PC-Dashboard	1165
PC-MASTERCONTROL	1351
PC-Menu	1810
PC-SuperMenu Plus	1975, 1976
Potpourri Menuing Program	0919
Pro-Menu	0798
ProMan	0897
Pushbutton	1041
Rokmar Floppy Disk Menu System	1739
SIMS	0895
Software Manager	1828
SuprMenu	2053
Window Menu	1637
YourMenu	1638

Printer Managers

Bradford	1053
Disk Spool II	0609
Epeat	1370
FXmaster	1369
Garc's Utilities	1370
LQ Printer Utility	0718
Prn Set	1303
ProMenu32	0779
SetPrint	1312
TPOP	1026
Typerite	0860
Versa-Spool	1606

Printer Utilities

Citizen Printer Utilities	1543
---------------------------	------

FXMatrix	0485
Letrhead	1519
Multi-Print	1661
Print Control Program	1522
Printer	0186
Printer Utilities 2	0326
Printer Utilities 4	0411
Printer Utilities 8	0438
Printer Utilities 7	1069
PRN-Test	1522
Rstprint	1662
Utilities Galore	1299
Warp-Ten and Print-matic	2048

Printer Utilities, Laser

2Faced	1487
A4Print	1079
Badge-O-Magic	2034
Dear Teacher HP Laser Font	1228
Download	1769
Font „Tiles“	2028
HP DeskJet Softfonts & Utilities	1462
HP LaserJet Soft Fonts and Utilities	1463
Laser Letterhead Plus	2037
Laser Printer Utility	2028
LaserEnvelope Printer	2028
P4UP	1460
Phamphiet	1222
Tiny Fonts for HP Laser Jet	1224
TSR Download	2031

Printing, Sideways

ON-Side	1184, 7073
Side Writer	0523

Security – File, Disk, or System

b-Crypt	1382
CLEAN-UP	2095
DosLock	1698
Encode/Decode	0482
EZ-CRYPT-LITE	1771
File-Safe	1993
FLU-SHOT+ and More	1119, 7087
LOCKTITE	1682
Login	1146
Memoirs5 Diary	0756
Mt-Log	1494
PC-Cryp2	0709
PC-Encrypt	1533
PC-Iris	2004
PC-Lock	2004
PC-Merlin	2004
Protect and Unprotect	0414
SCANRES	2095
Security Erase	1402
The Guardian	1787
VBug	2012
Virus Checker	1785
Viruscan	2095
Viruschk	1279

Utilities, Macro Programs

Function Key Helper	0856
Hotkey	0444
NEWKEY	0181

Utilities, System or Hardware

4DOS	1773
A400	0874
ANARKEY	1877
AT-Slow	1834
Burn-In	1335
Capacity	1682
Clock	0875
CloneRom	1408
CMOS-RAM/CHEKCMOS	1781
Date and Time	0908
DiskTest	2019

Drivechk & Align	1087
EGA Screen Save	0789
EGAbiank	0978
Fansi-Console	0356, 0650
gBlink	2029
Hdtest	1209
Hercules	0641
HGCIBM	0870
MakeSYS	1994
Maxi Form	1503
Montage2	0789
Patch	0779
PC-Kwick	1037
PC-Zipper	1572
PDVIM	0962
Procon Utilities	1537
Reconfig	1218
Reconflg	1037
ScreenSaver	1777
SCRNOFF3 & KYLOCK	0779
Speech	1377
Speech	1668
Swap Shop	0887
Test Drive	0908
The SWAP Utilities	1883
Vector	0874
VMS40	1957
ZZap	0874

Windows Application

Applications for Microsoft Windows	1915
Colorfix	1663
Fireworks	1663
GCP	1663

WORD PROCESSING & WRITING AIDS

Typing, Education

PC-FASTYPE	0320
PC-TOUCH	0249
Touch Type Tutor	1334
Typing	0793

Word Processor and Text Utilities

SS1 (text, db, spreadsheet)	1414, 1415
Alchemy Desktop Publishing Utilities	1360
AnyWord	1200
Armada Utilities	1376
Asc2com	1948
AVScripter	1028
BIBLIO	0935
Book Memo Display	1807
Breaker	0978
CAPBUF	1958
CleanUp	1818
CONGRESS-PCW	1821
CONGRESS-WP	1822
CopyFit-It	0898
DCA Conversion for PC-Write	1295
Do-It-Yourself Legal Forms	1949
Doc Master	0950
DocuHelp	1820
Door	1791
Dovetail III	0877
EasyType	1664
Elray Software Legal Disk Dictionary	1576
Every Other Page	1428
Executive Editor	1829
EZCount	0724
File Browser	1398
Grab Plus	1145
Hexcalibur	1782
LetterWriter	0719
MiniMax	1829
MMSOFT & MMREPORT DOCS	1868
MSPANTOC	0935
Multi-Merge	1152

Neat Text Formatter	1540
ParaSort	1858
PC-BROWSE	1670, 1671
PC-Write Font Selector	1267
PC-Write Macros	1457
PC-WRITE Pagemaker Import Filter	1267
Pen Pal	1570
PROINDEX	0977
QHELP/QTEXT	1853
Quattro & 1-2-3 to WordPerfect	2002
Readit	1687
REFLIST	0231
SeekEasy	0820
Shuffles	1312
Stripper	0978
SXU	0962
Text/File Handling Utility	1404
THE EDITING KEYPADS	1574
Thesaur – A Thesaurus program	1245
Thesaur Plus	1245
Word Processing Preview System	0415
WordStar 2000 Mouse Menus	1416
Writer's Heaven	0759

Word Processors, Text Editors, and Outliners

BlackBeard	0611
CopyCon	2029
EI	1615
E88 Text Editor	1102
ESL Writer	0899, 0900
FILEI	1893
Freeword	1084
Galaxy	0765
Idea Tree	1888
Jove	1429, 1430, 1454
KEDITOR	1643, 1665
MEGA-STAR	1766
MICRO-EMACS	1431, 1455
New York Edit	0829
New York Word	0528
PC-Ted	0898
PC-Type II	0455, 0681, 0682
PC-Write	0078, 0627, 1235, 7050
PC-Write Lite	2090, 2091
Phoenix Word	1770
QEdit Advanced	1783
QIP - The Quick Information Proc.	1477-80
RGB*Techwriter	1368
SageWords	1170
Softkey Editor	0880
The Wordsmith Newsletter Kit	1974
Visual Display Editor (VDE)	1273
W-ED	0415
WORD Fugue	1970, 1971, 2118
WordMaster	1686

Wordprocessors, Education

WordPerfect 5.0 Learning Syst.	1260, 1261
WordPerfect 5.0 Learning Syst.	1338, 1339

Writing and Composition Aids

A1001 Topics for Composition	1088
Basic English	1181
Book Report	1088
Cliche Finder	1525
Creativity Package	2075-7
Form Letters	0388
Handwriting Analyst	1050
invent	1181
Mastery Learning: Composition	1847-50
Names	1102
PC-Style (unsupported)	
Postease	1181
Poetry Generator	1166
Styled/Stylist	1181
THS English Tutorial	1742, 1743

POPIS NĚKTERÝCH VOLNĚ ŠÍŘENÝCH PROGRAMŮ

EDICE FCC PUBLIC

Témata zatím vydaných nebo připravovaných disket

- | | |
|---|---|
| #01 - programovací jazyk XLISP
- v prodeji, | #11 - počítačová grafická hra
MOSAIX, |
| #02 - emulace koprocessorů
8087, 80387 - v prodeji, | #12 - pakovací program PKPAK,
PKUNPAK a PKSFX, |
| #03 - knihovna č.1 pro jazyk
Turbo C, | #13 - speciální kopírovací pro-
gram DUPLICATOR, |
| #04 - knihovna č.1 pro jazyk
Turbo Pascal, | #14 - komunikační program
PROCOMM, |
| #05 - počítačová hra
CAPTAIN COMIC, | #15 - databázový program
WAMPUM, |
| #06 - programy HAM pro
radioamatéry, | #16 - kreslicí program pro děti
KIDPAINT, |
| #07 - překladač jednočipů 8048
a 8051, | #17 - knihovna č.1 pro progra-
mování v BASICu, |
| #08 - TASM - překladač proces-
orů Intel, Motorola, Zilog, | #18 - grafické simulátory ke
kartě HERCULES, |
| #09 - CAD program
DANCAD3D, | #19 - vzdělávací program
NIGHTSKY, |
| #10 - program pro výběr z menu
- NAVIGATOR, | #20 - antivirové programy SCAN
a CLEAN. |

PseudoSam Cross Assembler 68 a 685

Autorská verze : 1.4.x

*Autorská registrace : \$30.00 pro o-
sobní užívání, \$100.00 pro komerční
využití*

Speciální požadavky na hardware :
žádné

PSEUDOSAM 68 a 685 jsou křížové
překladače programů psaných v as-
sembleru procesorů typu Motorola
6800 a 6805. Tyto překladače Vám u-
možní psát a odlaďovat programy pro
tyto procesory na Vašem osobním po-
čítači PC-XT/AT.

PSEUDOSAM (znamená Pseudo-
brand Symbolic Assembler) podpo-
ruje jazykovou syntaxi, která vychází
z assembleru na UNIX System V.

Instrukce a adresová syntaxe odpo-
ovídá předpisům výrobce, ale direktivy
překladače, výrazy operandů a návěští
se liší.

Autoři zvolili tuto variantu syntaxe
kvůli velké popularitě operačního sys-
tému UNIX a také aby se vyhnuli pro-
blémům s udržením jisté kompatibility
s OEM assemblymi jiných výrobců.

Seznam souborů na disketě:

A68 .EXE
(překladač kódu procesoru 6800)

EXAMPLE .ASM
(příklady zdrojového textu)

LABGEN .ASM

MNEMTEST.ASM

SYN .ASM

A685 .EXE
(překladač kódu procesoru 6805)

EXAMPLE .ASM
(příklady zdrojového textu)

LABGEN .ASM

MNEMTEST.ASM

SYN .ASM

T1.ASM

TEST.ASM

PseudoSam Cross Assembler 18 a 65

Autorská verze : 1.4.x

*Autorská registrace : \$30.00 pro o-
sobní, \$100.00 pro komerční využití*

Speciální požadavky na hardware :
žádné

PSEUDOSAM 18 a 65 jsou křížové
překladače programů psaných v as-
sembleru procesorů RCA 1802, 1805
a Rockwell 6502. Tyto překladače Vám
umožní psát a odlaďovat programy
pro tyto procesory na Vašem osobním
počítači PC-XT/AT.

PSEUDOSAM (znamená Pseudo-
brand Symbolic Assembler) podpo-
ruje jazykovou syntaxi, která vychází
z assembleru na UNIX System V.

Instrukce a adresová syntaxe odpo-
ovídá předpisům výrobce, ale direktivy
překladače, výrazy operandů a návěští
se liší.

Autoři zvolili tuto variantu syntaxe
kvůli velké popularitě operačního sys-
tému UNIX a také aby se vyhnuli pro-
blémům s udržením jisté kompatibility
s OEM assemblymi jiných výrobců.

Seznam souborů na disketě

A18 .EXE (překladač)

EXAMPLE .ASM
(příklady zdrojového textu)

LABGEN .ASM

MNEMLEV1.ASM

MNEMLEV2.ASM

SYN .ASM

A65 .EXE (překladač)

EXAMPLE .ASM
(příklady zdrojového textu)

LABGEN .ASM

MNEMTEST.ASM

SYN .ASM

MAC .ASM

PseudoSam Cross Assembler 48 a 51

Autorská verze : 1.4.x

*Autorská registrace : \$30.00 pro
osobní užívání, \$100.00 pro komerční
využití*

Speciální požadavky na hardware :
žádné

PSEUDOSAM 48 a 51 jsou křížové
překladače programů psaných v as-
sembleru jednočipových procesorů
Intel 8748 a 8751. Tyto překladače Vám
umožní psát a odlaďovat programy
pro tyto procesory na Vašem osobním
počítači PC-XT/AT.

PSEUDOSAM (znamená Pseudo-
brand Symbolic Assembler) podpo-
ruje jazykovou syntaxi, která vychází
z assembleru na UNIX System V.

Instrukce a adresová syntaxe odpo-
ovídá předpisům výrobce, ale direktivy
překladače, výrazy operandů a návěští
se liší.

Autoři zvolili tuto variantu syntaxe kvůli velké popularitě operačního systému UNIX a také aby se vyhnuli problémům s udržení jisté kompatibility s OEM assembly jinych výrobců.

Seznam souborů na disketě:

A48.EXE
(překladač kódu procesoru 8748)

EXAMPLE.ASM
(příklady zdrojového textu)

LABGEN.ASM

MNEMTEST.ASM

SYN.ASM

TEST.ASM

A51.EXE
(překladač kódu procesoru 8751)

EXAMPLE.ASM
(příklady zdrojového textu)

LABGEN.ASM

MNEMTEST.ASM

SYN.ASM

TASM - tabulkový překladač assembleru

Autor: Thomas Anderson, Speech Technology Incorporated, 837 Front Street South, Issaquah, WA 98027

Autorská verze: 2.7.4

Autorská registrace: \$30.00

Speciální požadavky na hardware: žádné

TASM je speciální překladač assembleru pro počítače typu PC XT/AT, který umožňuje překládat zdrojové texty programů, určených pro osmibitové procesory: 8048, 8051, 8085, Z80, 6805 a 6502.

Seznam souborů na disketě

TASM.EXE
(hlavní program TASM)

TASM48.TAB
(tabulka instrukcí procesoru 8048)

TASM51.TAB
(tabulka instrukcí procesoru 8051)

TASM65.TAB
(tabulka instrukcí procesoru 6502)

TASM85.TAB
(tabulka instrukcí procesoru 8085)

TASM80.TAB
(tabulka instrukcí procesoru Z80)

TASM05.TAB
(tabulka instrukcí procesoru 6805)

TASM68.TAB
(tabulka instrukcí procesoru 6800/6801)

TASM32.TAB
(tabulka instrukcí procesoru TMS320)

TASM.DOC
(dokumentace k programu)

README
(vysvětlení k obsahu disku)

COPYRIGHT
(poznámka o ochranných právech)

ORDER.FRM
(formulář k registraci)

Uživatel může kterýkoliv z těchto souborů volně kopírovat a zkoušet. Chce-li však používat překladač pro svou činnost, je povinen zaplatit registrační poplatek!

Následující soubory může kopírovat pro svoji potřebu pouze vlastník registrace:

TASM.C
(zdrojový kód programu TASM)

MACRO.C
(zdrojový kód knihovny k programu)

PARSE.C
(zdrojový kód knihovny k programu)

TASM.H
(soubor k definici konstant programu)

TASM.MAK
(Make file k překladu programu TASM)

SIMCGA - simulace CGA na kartě HERCULES

Autor: Chuck Guzls, 153 North Murphy Ave., Sunnyvale, CA 94086

Speciální požadavky na hardware: grafická karta HERCULES

SIMCGA je jeden z řady simulátorů grafické karty CGA na kartě HERCULES. Je to speciální rezidentní program, který umožňuje používat programové produkty, vyžadující grafickou kartu CGA, i na kartě HERCULES (nebo kompatibilní). Grafika je pomocí simulátoru reprodukována na co největší možné ploše v normálním rastrovém poměru. Trik spočívá v tom, že karta HGC se programuje tak, aby za určitý čas zobrazila více řádků než odpovídá rozlišení CGA.

Zde použita metoda simulace pracuje celkem spolehlivě, přesto se právě proto, že jde jen o simulaci, objevuje několik drobných nedostatků, které není možné programově ovlivnit:

- Operace kopírování řádků se v reálném čase prolíná se zobrazováním, proto se u pohybujících se obrázků

nevyhnutelně musí objevovat „střiny“.

- Protože obrázek je při simulaci „expandován“ ve vertikálním směru na dvojnásobek, některé detaily jsou deformovány.
- Ty programy, které přímo čtou z registrů karty CGA na adrese 03d4, nenaleznou žádné registry ke čtení. Tato situace může způsobit havárii systému, naštěstí však takových programů není mnoho.
- Textový mód 40x25 není podporován.
- Pracujete-li v textovém módu (módy 0,1,2,3) standardní atributy monochromatické obrazovky nefungují.

Seznam souborů na disketě:

SIMCGA.COM
(Iniciální část programu)

SETCGA.COM
(zapnutí emulačního módu)

SETMONO.COM
(vypnutí emulačního módu)

SIMCGA.DOC
(dokumentace k programu)

Upozornění:

Jestliže simulátor používáte, nezapomeňte, že jakmile jej jednou spustíte, zůstává rezidentně v operační paměti až do příštího natažení systému. Programy SETCGA.COM a SETMONO.COM pouze zapínají nebo vypínají emulaci. Nedovedou však program z paměti odstranit!

Dobrodružství kapitána Comica (planeta smrti)

Autor: Michael A. Denio, 3106 Twinoaks Drive, Joliet, IL 60435

Autorská verze: 1.0

Autorská registrace: \$10.00 - 20.00

Speciální požadavky na hardware: EGA karta s 256kB pamětí

Dobrodružství kapitána Comica je hra s vysokou kvalitou barevné grafiky. U hráčů tato hra rozvíjí vynikajícím způsobem postřeh a rychlost rozhodování. Kapitán má za úkol objevit tři poklady na planetě Omsoc, které byly uloupeny a ukryty na neznámém místě v daleké zemi Tambl.

Během cesty musí kapitán, a vy s ním, čelit mnoha nebezpečím a nástrahám, které se snaží zabránit splnění jeho úkolu. Jsou to například: střemhlavé nálety ptáků, jedovatý hmyz, ohnivé koule, míče, pavouci, ropuchy, jiskry, talíře, kosmický prach, a další. Ale hrdinství kapitána nezná překážku, která by ho zastavila. Musí se často bránit a unikat smrtelným dotykům.

Přitom udlivuje svou hbitostí a obrovskými skoky. Ale nejenom nebezpečí číhá na každém kroku - na své pouti nachází kapitán i předměty, které mu alespoň trochu pomáhají. Jsou to klíče, lucerny, vývrtky, boty a plechovky od coca-coly.

Jestliže se rozhodnete vydat se společně s kapitánem Comicem na dobrodružnou výpravu, nutně budete potřebovat rychlé reflexy a dobrý zrak na detaily. Další věcí, kterou nesmíte zapomenout, je schopnost řešit problémy doslova a do písmene „za letu“. Tato hra je vhodná pro děti od sedmi let, ale nepochybuji o tom, že k ní zasednou i seriózní otcové v nejlepších letech.

Stručné instrukce

(stručné proto, že podrobný manuál je dodáván společně s touto hrou na jedné disketě)

Tento program je distribuován jako shareware a je určen každému, kdo ovládá triviální základy práce s počítačem. Pokud budete hrát Kapitána Comica, píše se v dokumentaci, měli byste zaplatit registrační poplatek deset až dvacet dolarů. Registrovaní uživatelé budou první, kdo dostanou další, vylepšenou verzi programu.

Spuštění

- tento program je spustitelný na všech počítačích IBM-PC XT/AT a kompatibilních s monitorem EGA s 256 kB video pamětí,
- po startu programu (COMIC + ENTER) je možno definovat vlastní klávesy na ovládání hry,
- po úvodních obrázcích můžete začít hrát.

Vlastní hra

- na začátku má kapitán pět životů a u sebe nemá žádný z předmětů, které mu ulehčují cestu,
- kapitán používá dva druhy energie:
 - a) ke střelbě - snižuje se frekvencí střelby,
 - b) ke své ochraně - ubývá po dotyku nepřátel,
- hra končí, jestliže kapitán splní svůj úkol (získá všechny tři poklady) nebo ztratí všechny své životy.

Pokyny

- kapitán má projít osm rozličných oblastí na Tambl - snaží se nalézt takovou cestu, která pro něj bude nejspokladnější,
- cestou se snaží co nejméně přijít do kontaktu s nepřáteli a kromě pokladů sbírat i předměty, které mu pomáhají,
- štít - vrací Comicovi část energie na obranu,
- plechovka od coly - zvyšuje Comicovu schopnost střelby,
- klíč - umožňuje otevírat dveře v různých oblastech,
- boty - zvětšují maximální délku skoků, atd.

Super Pinball

Autorská verze : 1

Autorská registrace : \$10.00

Speciální požadavky na hardware : CGA

Super Pinball je kolekce pěti velkých pinballových her. Ovládání je velice jednoduché a je zvoleno tak, aby co nejvíce odpovídalo skutečnosti a hráč měl pocit, že hraje na skutečném automatu kdesi v herně. Místo tlačítek na boku se páčky ovládají pravou a levou klávesou SHIFT. Mezi jednotlivými hrami je možnost návratu do DOSu. Tato disketa patří k těm, které jsou dodávány bez dokumentace - není jí totiž třeba.

Seznam souborů na disketě:

BLACKBOX.COM

CHUTE.COM

DOUBLE.COM

STAR.COM

ZIPPER.COM

Slot

Autorská verze : 1.0

Autorská registrace : \$10.00

Speciální požadavky na hardware : EGA

Slot (od Ted Gruber Software) je věrná napodobenina hracího automatu, který se pomalu začíná rozšiřovat po celé republice. Jeho počítačová verze má jednu výhodu a jednu nevýhodu. Výhoda - při této hazardní hře na svém počítači neutratíte ani korunu (přesněji ani dolar). Nevýhoda - žádná mínce vám však z vašeho computeru nevypadne, ani když budete vysoko vyhrávat. Tato hra je spustitelná na počítačích s EGA kartou.

Seznam souborů na disketě:

SLOT.EXE (vlastní hra)

SLOT0.PMB (datový soubor)

SLOT1.PMB (datový soubor)

MOSAIX

Autorská verze : 1.0

Autorská registrace : \$35.00

Adresa : PC-SIG, 1030D East Duane Avenue, Sunnyvale Ca. 94086, (408) 730-9291, tel. (614) 888-8088

Speciální požadavky na hardware : 386kB RAM, CGA, EGA, VGA, dvě diskové mechaniky nebo pevný disk.

MOSAIX je vlastně, jak už název napovídá, počítačová skládanka. Obrázky, které se svou kvalitou vyrovnají barevné fotografie, jsou podle obtížnosti rozděleny na 25, 50 nebo 100 stejně velkých dílků. Ty jsou potom náhodně rozmístěny po obrazovce a úkolem je složit původní obrázek. Ve verzi, kterou nabízíme, jsou tři. Ale je možné vytvořit další například pomocí digitizéru z videa, z barevného či černobílého scanneru, anebo je možné použít nějaký kreslicí program. MOSAIX je však schopen přijmout pouze grafický formát PCX. Pokud pošlete na výše uvedenou adresu s registračním poplatkem ještě navíc \$10.00, bude vám zaslána vylepšená verze a navíc program ARGUS, který je schopen převést jakýkoli formát (např. CUT - Halo I-II-III, Dr. Halo; IMG - GEM Paint; GIF; PIC; TIF ...) do grafického formátu PCX. MOSAIX je doporučeno používat na grafické kartě VGA, MCGA nebo EGA (je možno použít i CGA kartu, ale grafická knihovna této karty není dodávána). MOSAIX lze ovládat pomocí klávesnice, myši nebo křížovým ovladačem. Způsob instalace, spuštění, návod a další informace jsou uvedeny v desetistránkové připojené dokumentaci.

The Night Sky

Autor : A. C. Stevely, 5c, Saint Johns Vale, London SE8 4EA, tel. 01 692 8265

Autorská verze : 2.06

Autorská registrace : 14.95 liber šterlinků

Speciální požadavky na hardware : Hercules nebo CGA

V podstatě se jedná o počítačové planetárium, které je schopno zobrazit až 2 900 hvězd do velikosti +5,49 a 8 500 hvězd do velikosti +6,49. Jde o produkt uživatelsky velice jednoduše ovladatelný pomocí menu, která vás dovedou přímo k požadované službě.

The Night Sky kreslí hvězdnou mapu a různé přesuny po obloze, podle vašeho přání zvětšuje hvězdy a rozkládá souhvězdí, je schopen nakreslit všechny polohy planet a Měsíce v časovém intervalu od 1. ledna 1600 do 31. prosince 3000, předpovídá zatmění Slunce a Měsíce, ukazuje místní oblohu. Dále obsahuje informace o kometách a hvězdách trpasličích, Slunci, Měsíci a satelitech Jupitera.

Uživatel si může nastavit souřadnice vlastní polohy, rok, měsíc, den a čas, a má na obrazovce hvězdnou oblohu tak, jak by ji měl za jasné noci

nad hlavou venku. To vše doma, vřep-
le...

Na konci dokumentace je uveden seznam literatury. Z tohoto seznamu i ze skutečnosti, že astronomie je autorovým koníčkem, je jednoznačně patrné, že program není udělán pro lačný efekt ale pro poctivou a přesnou práci. A na úplném konci autor uvádí, že v brzké době začne pracovat na další verzi, která bude navíc obsahovat představy o hlubokém kosmu. Galaxie, která se vám objeví jako titulní obrázek, je příkladem jedné z nich. Samozřejmě registrovaný uživatel dostane tuto novější verzi zdarma.

Autora by zajímaly připomínky a poznámky, ke kterým dospějete při užívání tohoto programu.

Seznam souborů na disketách:

NIGHT.EXE	(start programu)
PROG1.TBC	(hvězdy)
PROG2.TBC	(planety)
PROG3.TBC	(zatmění)
PROG4.TBC	(místní obloha)
PROG5.TBC	(komety a hvězdy trpaslíci)
PROG6.TBC	(Slunce)
PROG7.TBC	(Měsíc)
PROG8.TBC	(satelity Jupitera)
HOME.DAT	(vaše poloha, data)
TITLE.DAT	(titulní obrázek)
MANUAL.DOC	(dokumentace)
STARNUM.DAT	(uložená data o hvězdné mapě)
STARMAP.DAT	(hvězdný atlas)
MAG5.DAT	(hvězdy velikosti od +3,50 do +5,49)
MAG3.DAT	(hvězdy velikosti od -1,50 do +3,49)
DEEPSKY.DAT	(objekty hlubokého kosmu)
DEEPSKY2.DAT	(více informací o objektech hlubokého kosmu)
CONSTEL1.DAT	(jména konstelací)
CONSTEL2.DAT	(text základních map konstelací)
CONSTEL3.DAT	(data pro kreslení drah)
BSTAR21.DAT	(21 nejsvětějších hvězd)
BRIGHT3.DAT	(katalog jasných hvězd)
READ.ME	(základní informace)

Home Movie Librarian

Autor : Jeff Thorness, 81-875 Ave 48
#148, Indio, CA 92201.

Autorská verze : 1.0

Autorská registrace : \$20.00

Speciální požadavky na hardware :
pevný disk

Home Movie Librarian je vlastně databázový program, pomocí kterého si můžete vytvářet databanku videokazet, které vlastníte. Katalog mimo jiné obsahuje název filmu, počáteční a koncovou pozici, délku filmu, druh videokazety, žánr, jména herců a režiséra, číslo kazety, atd.

Tímto způsobem lze vybírat filmy podle žánrů, typů kazet, herců atd. Jestliže máte navíc k počítači připojenou tiskárnu, veškeré výpisy je možno rovnou tisknout.

Seznam souborů na disketě:

HML.ARC	(zarchivovaný program)
READ.ME	(úvodní textový soubor)
PKXARC.COM	(rozpakovávací program)
HARD.BAT	(instalační dávkový soubor)

Kid Paint

Autorská verze : 3.004

Autorská registrace : \$4.50

Speciální požadavky na hardware :
EGA karta s 256kB pamětí, 2 diskové mechaniky nebo pevný disk

Adresa : COMPASS SYSTEMS,
P.O. Box 84, Hatboro, Pa 19040

Jedná se o dětskou omalovánku. Dítěti je předložen obrázek, který si může podle své fantazie libovolně vybarvovat. K dispozici má přes padesát barev a výplní. Ve verzi, kterou máme možnost Vám představit, jsou obrázky pouze dva, ale pokud pošlete na uvedenou adresu registrační poplatek a 17,50 dolaru (na diskety), dostanete od firmy verzi 4.0, kde je obrázků už osm (viz konec tohoto programu). Ovládání odpovídá samozřejmě věku dítěte a je velice jednoduché. Je možno vybrat si mezi klávesnicí a myší (dítě si se ní naučí rychle zacházet).

Seznam souborů na disketě:

KPAINT1.ZIP	
KPAINT2.ZIP	(programy ve zkomprimovaném tvaru)
PKUNZIP.EXE	(rozbalovací program)

KIDGAMES

Autor : Donald L. Pavla, 1488 Lahti Drive, Bellingham, WA 98226

Autorská registrace : nesmí být vyšší než \$10.00 (\$15.00 za zdrojový kód v Turbo Pascalu)

Speciální požadavky na hardware :
žádné

Tyto hry byly napsány pro autorovy děti. Na disku jsou následující soubory:

HANGMAN	(učí písmena a výslovnost, bližší informace jsou v souboru HANGMAN.DOC)
ALPHABET	(učí abecedu a pořadí v abecedě)
ANIMALS	(učí jednoduchou předškolní abecedu)
CLOCKGAME	(učí hodiny)
MOSAIC	(učí prostorovou představivost)

a další potřebné soubory.

Googol Math Games

Autorská verze : 2.0

Registrační poplatek : \$10.00

Speciální požadavky na hardware :
386kB RAM, CGA (nebo EGA/VGA)

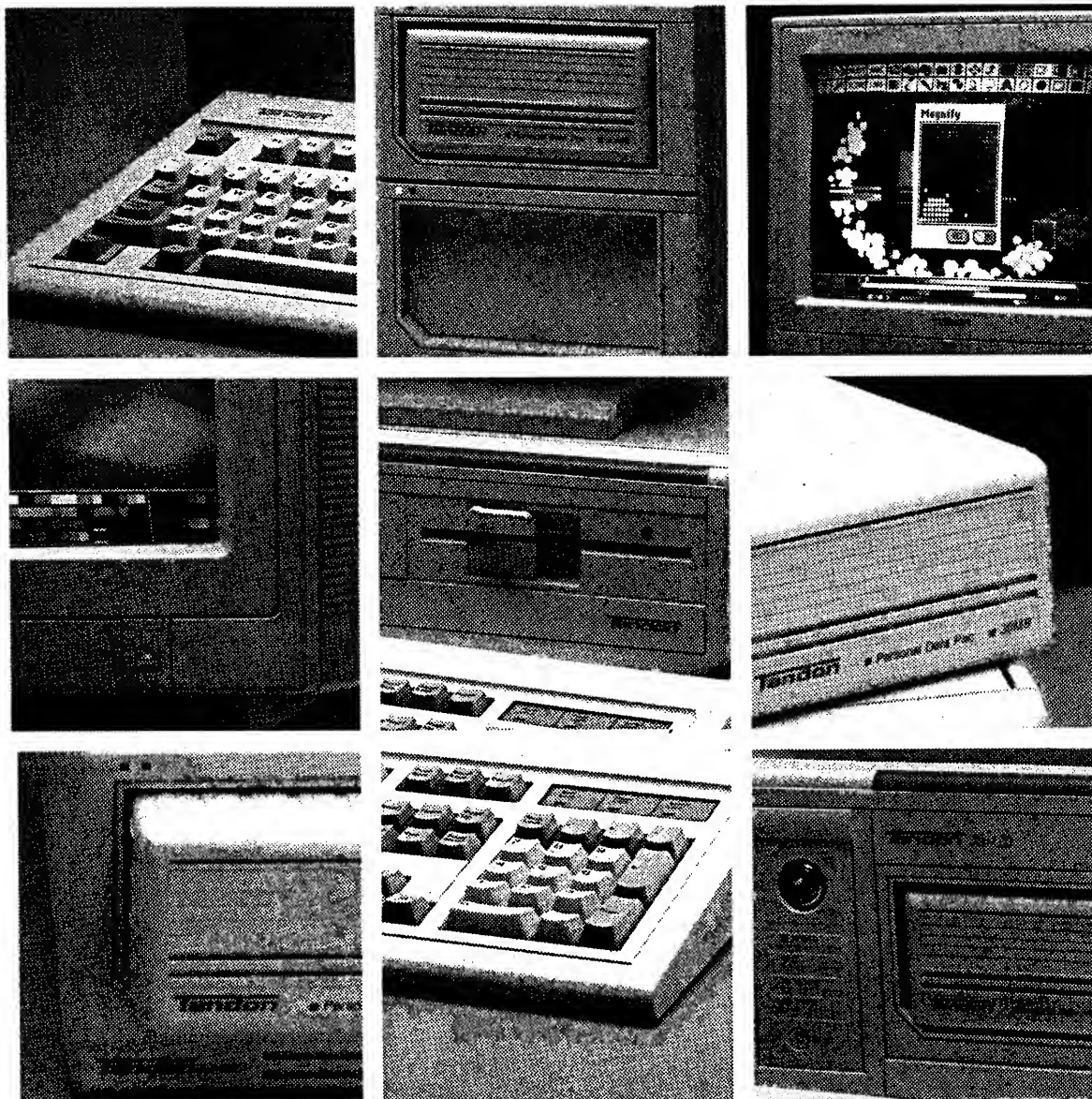
Vaše děti si jistě oblíbí tyto dobrodružné matematické hry. Jsou vytvořeny tak, že hráč, pokud chce vyhrávat, musí umět všechny základní matematické operace - tedy sčítání, odčítání, násobení a dělení. Jejich grafické řešení je navíc velmi zajímavé a poutavé, takže hrající si dítě netuší, že je vlastně žákem.

Tyto hry mohou hrát i velice malé děti. Pokud ještě neumějí číst, rodiče jim jistě rádi pomohou. Starší děti, pěti až desetileté, mohou celou hru ovládat samostatně.

MIMOŘÁDNÝ KUPÓN

VOLNÉ ŠÍŘENÉ PROGRAMY
FCC PUBLIC

Nalepíte-li tento kupón na Vaši objednávku volně šířených programů u firmy FCC Folprecht, dostanete na objednané programy 10% slevu!



Tato část ročenky Počítačová elektronika byla připravena ve spolupráci s firmou FCC Folprecht, spol. s r.o., Velká hradební 48, 400 01 Ústí nad Labem. Společně připravujeme i založení rubriky **Public Domain - volně šířené programy** v Amatérském radiu. Firma FCC Folprecht bude poskytovat potřebné podklady a zajišťovat zasílání programů všem, kteří si je objednají. V rubrice v AR vám chceme pomoci se trochu orientovat v nepřehledném sortimentu a množství volně šířených programů.

FILE OUT & FILE IN

PROGRAMY PRO OVLÁDÁNÍ VSTUPU A VÝSTUPU Z/DO TEXTOVÉHO SOUBORU

Pavel Kříž, Paláskova 1107, 182 00 Praha 8

Dvojice systémových programů FILE OUT a FILE IN umožňuje použít pro čtení nebo pro zápis textový soubor a tak rozšiřuje možnosti využití vstupu a výstupu na počítačích typu Sinclair ZX Spectrum.

Použití programu FILE OUT

Na počítači Spectrum je možné řídit výstup na TV obrazovku nebo na tiskárnu. Program FILE OUT rozšiřuje tyto možnosti o další - výstup do souboru.

Po přeměrování výstupu do souboru se znaky zpracovávají a zapisují do tohoto souboru. Při zobrazení v textovém editoru TASWORD nebo SPECTRAL WRITER je pak výpis takový, jaký by byl při přímém výstupu na obrazovku nebo na tiskárnu.

Výstup do textového souboru je výhodné použít z mnoha důvodů:

- Místo přímého tisku na tiskárnu si vytvoříte nejprve textový soubor. Ten si můžete v editoru prohlédnout, libovolně upravit a teprve potom vytisknout.
- Můžete provést výpisy programů v BASICu, HiSoft PASCALu, HiSoft C, Assembleru apod. Vzniklé textové soubory pak v editoru opatříte poznámkami, začleníte do textu apod.
- Při výpočtech se během chodu programu ukládají výsledky do souboru. Bloky souboru se nahrávají na magnetofon. Tam zůstanou výsledky uchované a pokud nejsou vypočtené hodnoty výchozími pro pokračování v dalším výpočtu, nemusíte se dále uchovávat v paměti počítače.
- Výhoda výstupu do souboru bude markantní zvláště u dlouhotrvajících výpočtů s průběžnými výpisy vypočtených hodnot, kdy nebudou kladeny žádné nároky na tiskárnu ani na obsluhu.
- Výsledky, které se zapsaly do souboru, můžeme využít pro čtení dat programem FILE IN (viz dále).

Kromě výše uvedených možností má program FILE OUT další nesporné výhody:

- Můžete kdykoliv vytisknout již jednou vytvořený soubor s výsledky bez opakování výpočtů.

- Soubor může být uložen na páse ve zkomprimovaném tvaru.
- Své programy můžete použít bez jakýchkoli úprav.

Použití programu FILE IN

Vstup je na počítači Spectrum možné přijímat Implicitně pouze z klávesnice. Program FILE IN rozšíří vstup o vstup z textového souboru. Místo zadávání z klávesnice se hodnoty načítají z tohoto souboru.

Vstup ze souboru použijete v následujících případech:

- V textovém editoru napíšete soubor dat a zaznamenáte ho na magnetofon. Z tohoto souboru se pak všechny hodnoty budou načítat programově.
- Použijete soubor dat, který vznikl použitím programu FILE OUT. Z něho pak je možné číst dříve zadané nebo vypočtené hodnoty.

Další výhody programu:

- FILE IN dokáže číst i ze zkomprimovaného souboru.
- Kombinací programů FILE OUT a FILE IN je dosaženo vysoce využitelné komunikace mezi různými programy.

Uvedené příklady použití programů není možné bez systému FILE OUT/IN realizovat vůbec nebo pouze s obtížemi.

POPIS PROGRAMU FILE OUT

Program FILE OUT (dále označovaný zkráceně FO) vytvoří po inicializaci v paměti počítače prázdný textový soubor, otevře ho pro zápis a přeměruje do něho výstup. Před otevřením souboru je možné definovat jeho velikost (počet řádek x počet sloupců) a umístění v paměti. Do souboru se zapisuje příkazem LPRINT, např.

LPRINT „text“, „cislo:“;alfa

Použitím příkazu OPEN # zajistíte, že je do souboru možné zapisovat i pří-

kazem PRINT. Užití příkazu OPEN # je popsáno v kapitole *Napojení programů FO a FI na systém*.

Pro zápis do souboru je možné použít stejné prostředky, jaké by byly použity pro tisk na obrazovku nebo ZX PRINTER. Program FO obsahuje řadu řídicích kódů, které tyto možnosti rozšiřují. Další řídicí kódy pak slouží pro ovládání programu FO.

Seznam řídicích kódů:

- Inicializace (nastavení Implicitních hodnot),
- nastavení délky strany,
- nastavení spodního okraje,
- nastavení levého okraje,
- nastavení pravého okraje,
- nastavení horizontálního tabelátoru,
- nastavení vertikálního tabelátoru,
- zadání názvu souboru,
- nastavení čísla bloku souboru,
- volba nahrávání po uzavření souboru,
- volba pauzy před nahráváním,
- volba komprimace před nahráváním,
- volba výpisu klíčových slov,
- volba zvukového signálu při chybě,
- volba výpisu otazníku při chybě,
- volba pohybu kurzoru na stránce,
- nastavení polohy kurzoru na stránce,
- nastavení polohy kurzoru na řádce,
- nová stránka,
- nová řádka,
- horizontální tabelátor,
- modifikovaný horizontální tabelátor (čárka),
- vertikální tabelátor,
- posunutí kurzoru o několik sloupců,
- posunutí kurzoru o několik řádků,
- zvukový signál (zvoneček),
- uzavření souboru.

Během tisku jsou rozlišovány sloupce, řádky a stránky bloku souboru. Na začátku každé stránky je řádek 0 a sloupec 0.

Celkový počet řádek je samozřejmě omezený (podle velikosti přidělené paměti). Když dojde k přeplnění, soubor se uzavře a dále se FO zachová rozdílně podle nastavení voleb.

Pokud je vypnuto nahrávání, při každém dalším pokusu o zápis do souboru se systém ohlásí chybovou zprávou

End of file.

Řídicí kódy, které nemění polohu kurzoru a provádějí pouze nastavování parametrů a voleb, se akceptují i po uzavření souboru.

Když je nahrávání zapnuto, vykonají se následující akce:

- zazní zvoneček,
- změní se barva okraje obrazovky na kontrastní s barvou původní,
- čekání v délce 1 s,
- při nastavení pauzy FO čeká na stisknutí libovolného tlačítka,

- nastaví se původní barva okraje obrazovky,
- při nastavení komprimace se soubor zkomprimuje,
- blok souboru se uloží do vnější paměti,
- Inkrementuje se číslo bloku souboru,
- soubor se vyprázdní a uvolní pro další zápis.

Soubor se nahrává ve formě „bytes“ s názvem zadáním pomocí jednoho z řídicích kódů. Bloky souboru se automaticky číslovají od 1 do 99 (a pak znovu od 0). Celý soubor je tedy rozdělen na bloky s čísly 01, 02, ..., nn, které tvoří vždy 9. a 10. znak názvu. Počáteční číslo je možné také nastavit jedním z řídicích kódů.

Ovládání programu FO

Před spuštěním programu FO je možné předefinovat umístění textového souboru v paměti počítače a jeho velikost = počet řádek x počet sloupců (viz příloha A). Implicitně se předpokládá umístění souboru v paměti od adresy 32000, počet řádků 320, počet sloupců 64, délka souboru je tedy $320 \times 64 = 20480$ bajtů a poslední adresa souboru $32000 + 20480 - 1 = 52479$.

Program FO je sestaven tak, že pracuje s libovolným počtem znaků na řádek, ovšem pro správné zobrazování v textových editorech TASSWORD nebo SPECTRAL WRITER je nutné ponechat 64 znaků na řádek.

Po definování velikosti souboru se program FO spustí, např.

RANDOMIZE USR cold

Tím se otevře pro zápis prázdný textový soubor.

Dále se již pro ovládání programu používají některé z řídicích kódů. Následuje jejich popis, ve kterém je použito:

n,l,c ... číslo v rozsahu dle typu kódu (max. 0 až 255),

x volba 0 (Ne), 1 (Ano).

Pokud by došlo k zadání parametru mimo povolený rozsah, nastavení se neprovede a ohlásí se tisková chyba.

Formát většiny řídicích kódů je vytvořen dle standardu EPSON ESC/P. Speciální řídicí kódy (neobsažené v ESC/P) jsou dále označeny *).

Inicializace, volby

Inicializace

ASCII: ESC @
BASIC: LPRINT CHR\$ 27;"@";

Provede nastavení implicitních hodnot:

- levý okraj = 0,
- pravý okraj = definovaný počet znaků/řádek (64),

- spodní okraj = 0,
- horizontální tabelátor = (definovaný počet znaků/řádek)/4 (16),
- vertikální tabelátor = 1,
- zapnuto,
- výpis klíčových slov,
- výpis otazníku při chybě,
- pohyb kurzoru na stránce,
- pauza před nahráváním,
- vypnuto,
- nahrávání,
- komprimace.

Zadání názvu souboru *)

ASCII: ESC \$...data... CR
BASIC: LPRINT CHR\$ 27;"\$.
data. ..."
Implicitně: "\$ " (prázdný název)

Název může mít až 8 znaků, další znaky se ignorují.

Zároveň se změnou názvu se automaticky nastaví počáteční číslo bloku souboru rovno jedné.

Pozor, název musí být ukončen znakem CR (ENTER).

Zadání čísla bloku souboru *)

ASCII: ESC n n
BASIC: LPRINT CHR\$ 27;"n";
CHR\$ n;
Implicitně: 1 pro počáteční blok souboru
Rozsah: 0 až 99

Tímto řídicím kódem nastavíte počáteční nebo aktuální číslo bloku souboru.

Volba nahrávání po uzavření souboru *)

ASCII: ESC s x
BASIC: LPRINT CHR\$ 27;"sx";
Implicitně: Ne

Volba pauzy před nahráváním *)

ASCII: ESC w x
BASIC: LPRINT CHR\$ 27;"wx";
Implicitně: Ano

Při nastavené pauze čeká FO vždy před nahráním bloku souboru na stisknutí libovolného tlačítka. Když není pauza nastavena, spustí se nahrávání po jedné sekundě.

Volba komprimace *)

ASCII: ESC c x
BASIC: LPRINT CHR\$ 27;"cx";
Implicitně: Ne

Při nastavené komprimaci tato proběhne automaticky před nahráním bloku souboru. Délka zkomprimovaného souboru je přibližně 2/3 původní délky. Při „řidším“ zápisu můžete očekávat zkrácení na 1/4 i méně.

Pro dekomprimaci je nutné použít editor Tasword Kompres.

Volba výpisu klíčových slov *)

ASCII: ESC t x
BASIC: LPRINT CHR\$ 27;"tx";
Implicitně: Ano

Při vypnutém výpisu klíčových slov se při pokusu o tisk znaku s kódem 165 až 255 zapíše do souboru otazník.

Volba zvukového signálu při chybě *)

ASCII: ESC b x
BASIC: LPRINT CHR\$ 27;"bx";
Implicitně: Ano

Tisková chyba nastává v těchto případech:

- znak mimo rozsah ASCII 32 až 143,
- neplatný řídicí kód,
- parametr řídicího kódu mimo povolený rozsah.

Když je zvukový signál nastaven, pak při výskytu tiskové chyby se FO ohlásí pípnutím, v běhu programu ale pokračuje.

Viz též další odstavec.

Volba výpisu otazníku při chybě *)

ASCII: ESC ? x
BASIC: LPRINT CHR\$ 27;"?x";
Implicitně: Ano

Při výskytu tiskové chyby (viz předchozí odstavec) zapíše FO do souboru otazník (pokud je nastaven).

Volba pohybu kurzoru na stránce *)

ASCII: ESC p x
BASIC: LPRINT CHR\$ 27;"px";
Implicitně: Ano

Pokud je volba nastavena, pak je možné příkazem AT přesunout kurzor na libovolné místo stránky (tedy i vracet zpět). Pokud chcete výpis na obrazovku změnit na tisk do souboru, použijte OPEN #2,"P" (viz kapitola *Napojení programů FO a FI na systém*) a pak u každého přechodu na novou stranu (CLS) přidejte do programu např. LPRINT CHR\$ 12;. Jinak by při provádění AT docházelo k prepisování textu. Tomu můžete zamezit vypnutím této volby, pak ale musíte soubor v editoru upravit.

Při vypnuté volbě se příkaz AT chová stejně jako příkaz TAB, přesunutí na určitý řádek se tedy neprovede.

Formátování strany

Nastavení délky strany

ASCII: ESC C n
BASIC: LPRINT CHR\$ 27;"C";
CHR\$ n;
Implicitně: MIN (80, def. počet řádků souboru)
Rozsah: 1 až MIN (80, def. počet řádků souboru)

Pokud se kurzor nenachází na začátku stránky, přesune se na začátek další stránky. Pak se teprve nastaví délka strany.

Nastavení spodního okraje

ASCII: ESC N n
BASIC: LPRINT CHR\$ 27;"N";
CHR\$ n;
Implicitně: 0
Rozsah: 0 až délka strany - 1

Spodní okraj určuje počet řádků od konce strany, do kterých se již nezapíše.

Nastavení levého okraje

ASCII: ESC I n
BASIC: LPRINT CHR\$ 27;"I";
CHR\$ n;
Implicitně: 0
Rozsah: 0 až pravý okraj - 1

Levý okraj stanoví počet prvních n sloupců, do kterých se nebude zapisovat. Veškerý další tisk se bude vztahovat k levému okraji, kde je dočasně platný sloupec 0.

Například při přechodu na nový řádek se kurzor přesune na pozici n sloupců od skutečného okraje nového řádku.

Nastavení pravého okraje

ASCII: ESC Q n
BASIC: LPRINT CHR\$ 27;"Q";
CHR\$ n;
nebo
ASCII: ESC r n *)
BASIC: LPRINT CHR\$ 27;"r";
CHR\$ n;
Implicitně: definovaný počet
znaků/řádek (64)
Rozsah: levý okraj + 1 až def.
počet znaků/řádek

Pravý okraj stanoví dočasný konec řádku a určuje se od skutečného levého okraje. Využitý počet sloupců na jednom řádku je tedy: pravý okraj - levý okraj.

Řízení kurzoru

Nastavení horizontálního tabelátoru

ASCII: ESC e 0 n
BASIC: LPRINT CHR\$ 27;"e0";
CHR\$ n;
Implicitně: (definovaný počet
zn./ř.) / 4 (16)
Rozsah: 0 až 32

Parametr bude rozhodující při výpočtu tabulačních pozic, které jsou na sloupcích 0,n,2*n,3*n,...

Při n=0 se bude kód HT pro tabulaci ignorovat.

Nastavení vertikálního tabelátoru

ASCII: ESC e i n

BASIC: LPRINT CHR\$ 27;"ei";
CHR\$ n;
Implicitně: 1
Rozsah: 0 až délka strany

Parametr je použit pro výpočet tabulačních pozic, které jsou na řádcích 0,n,2*n,3*n,...

Při n=0 se bude kód VT pro tabulaci ignorovat.

Posunutí o několik sloupců

ASCII: ESC f 0 n
BASIC: LPRINT CHR\$ 27;"f0";
CHR\$ n;
Rozsah: 0 až 255

Pokud se překročí dosud platný řádek, přesune se kurzor na začátek příštího řádku a další posunutí již neproběhne.

Při n=0 bude posunutí ignorováno.

Posunutí o několik řádků

ASCII: ESC f 1 n
BASIC: LPRINT CHR\$ 27;"f1";
CHR\$ n;
Rozsah: 0 až 255

Když se překročí dosud platná stránka, přesune se kurzor na začátek příští stránky a další posouvání se již neprovádí.

Při posouvání zůstane zachován dosud platný sloupec.

Pro n=0 bude posunutí ignorováno.

Nová stránka

ASCII: FF
BASIC: LPRINT CHR\$ 12;

Kurzor se přemístí na začátek další stránky.

Nová řádka

ASCII: LF
BASIC: LPRINT CHR\$ 10;
nebo
ASCII: CR
BASIC: LPRINT CHR\$ 13;
nebo LPRINT
nebo LPRINT

Kurzor se přemístí na začátek další řádky.

Horizontální tabelátor

ASCII: HT
BASIC: LPRINT CHR\$ 9;

Kurzor se přesune na nejbližší vyšší tabulační pozici. Pokud by se tím překročil dosud platný řádek, bude se tabulace ignorovat.

Modifikovaný horizont. tabelátor *)

ASCII: ACK
BASIC: LPRINT CHR\$ 6;
nebo LPRINT

Stejně jako výše, ale když má dojít k překročení dosud platného řádku, přesune se kurzor na začátek dalšího řádku.

Vertikální tabelátor

ASCII: VT
BASIC: LPRINT CHR\$ 11;

Kurzor se přesune na začátek řádku nejbližší tabulační pozice. Pokud by se překročila dosud platná stránka, přesune se kurzor na začátek další stránky.

Nastavení polohy kurzoru na stránce *)

ASCII: SYN I c
BASIC: LPRINT CHR\$22;
CHR\$I; CHR\$c;
LPRINT AT I,c;
nebo
Rozsah: I: 0 až délka stránky -
spodní okraj - 1
c: 0 až pravý okraj -
levý okraj - 1

Kurzor přeskočí na řádek I a sloupec c platné stránky. Toto je jediný řídicí kód, který umožňuje návrat kurzoru zpět, proti směru tisku.

Viz též volba pohybu kurzoru na stránce.

Nastavení polohy kurzoru na řádce *)

ASCII: ETB c NUL
BASIC: LPRINT CHR\$ 23;CHR\$c;CHR\$ 0;
nebo LPRINT TAB c;
Rozsah: 0 až pravý okraj - levý okraj - 1

Pokud by se kurzor musel na řádce vracet, přemístí se na sloupec c další řádky.

Ostatní

Zvoneček

ASCII: BEL
BASIC: LPRINT CHR\$ 7;

Zazní zvukový signál odpovídající BASICovému BEEP 0.1,24 s prodlevou 0.1 s (PAUSE 5).

Uzavření souboru *)

ASCII: ESC \
BASIC: LPRINT CHR\$ 27;"\";

Uzavře se soubor. To je vhodné použít po ukončení tisku pro nahrání posledního bloku souboru.

Program FILE OUT je vytvořen ve dvou verzích:

1. file-out.l
umístění 30000 až 31357
2. file-out.h
umístění 64000 až 65357

Dále použijte hodnoty podle verze:

verze	file-out.l	file-out.h
start cold	30000	64000
start warm	30003	64003
TOP	31313	65313
N.LIN	31315	65315
N.COL	31317	65317

Pro změnu dvoubajtových hodnot použijte funkce:

DEF FN L(x)=x-256*FN H(x)

DEF FN H(x)=INT(x/256)

Definice umístění souboru v paměti počítače:

Změňte dvoubajtovou hodnotu TOP:

POKE TOP, FN L(umístění)

POKE (TOP+1), FN H(umístění)

Implicitní hodnota umístění souboru je 32000.

Definice velikosti souboru:

Počet řádků: změňte dvoubajtovou hodnotu N.LIN

POKE N.LIN, FN L(řádků)

POKE (N.LIN+1), FN H(řádků)

Počet sloupců: změňte jednobajtovou hodnotu N.COL

POKE N.COL, sloupců

Implicitní velikost souboru je 320 řádků x 64 sloupců.

Definice umístění a velikosti souboru je možné použít pouze před spuštěním, pak se již tyto hodnoty nemohou měnit (až do opětovného spuštění).

Spuštění programu:

RANDOMIZE USR cold

1. vyprázdní předem definovaný soubor (vypíná mezery),
2. nastaví výstup do tohoto souboru,
3. nastaví všechny implicitní hodnoty,
4. nastaví kurzor na začátek souboru.

RANDOMIZE USR warm

2. - 4. jako u cold.

Příklad použití FO

1. tabulka funkce sinus

10 PRINT "Tabulka", "u", "SIN (u)"

20 PRINT "-----"

30 FOR u=0 TO 360 STEP 10

40 PRINT u, SIN (u*PI/180)

50 NEXT u

Výpis přesměrujete do souboru použitím **OPEN #2, "p"**.

2. tabulka malé násobilky

10 LPRINT CHR\$ 27;"e0";CHR\$ 5;:

REM nast. horiz. tab.

20 LPRINT TAB 21;"Tabulka male nasobilky"

30 LPRINT ' , ,

40 FOR i=1 TO 10

50 LPRINT i,

60 NEXT i

70 LPRINT

80 FOR j=1 TO 10

90 LPRINT ' j , ,

100 FOR i=1 TO 10

110 LPRINT i*j,

120 NEXT i

130 NEXT j

Nejlépe bude, vyzkoušíte-li výsledek přímo v TASWORDu. K tomu použijte následující postup:

1) nahrajte program TASWORD:

LOAD "tasword"

2) proveďte návrat do BASICu:

(STOP, B, ENTER)

3) vymažte basicový program:

NEW

4) snižte RAMTOP:

CLEAR 29999

5) nahrajte program FO:

LOAD "file-out.l" CODE 30000

6) inicializujte FO:

RAND USR 30000

7) napište program (ukázka 1, 2):

8) a spusťte ho:

RUN

9) nyní se můžete podívat, co je v souboru:

RAND USR 64330

tím spustíte TASWORD.

3. výpis programu do souboru

Napište nebo nahrajte nějaký basicový program. Nahrajte program FO.

Nyní proveďte:

1) nastavení volby nahrávání:

LPRINT CHR\$ 27; "s1";

2) zadání názvu souboru:

LPRINT CHR\$ @&; "SProgram"

3) výpis programu:

LLIST

4) uzavření souboru:

LPRINT CHR\$ 27; "\";

5) spuštění nahrávání:

LPRINT

Po stisknutí tlačítka proběhne SAVE.

POPIS PROGRAMU FILE IN

Program FILE IN (dále označovaný zkráceně FI) po inicializaci otevře pro čtení textový soubor a přesměruje z něho vstup. Před otevřením souboru je možné definovat jeho velikost (počet

řádků x počet sloupců) a umístění v paměti.

Po definování velikosti souboru se program FI spustí, např.

RANDOMIZE USR cold

Tím se vyprázdní a otevře textový soubor pro čtení.

Data jsou uvnitř souboru od ostatního textu oddělena ohraničením složenými závorkami. Jednotlivé hodnoty nebo výrazy jsou oddělené mezerou.

Ze souboru se čte příkazem **INPUT #3 (případně INKEY\$#3)**, např.

INPUT #3;A;B;C;R\$;

Program FI obsahuje čítač znaků, který se posunuje od čísla k číslu. Po načtení prvního čísla se čítač posune na druhé atd. Pokud dojde k překročení délky souboru (tedy pokus o načtení více čísel, než kolik jich soubor obsahuje), zachová se FI rozdílně podle nastavené volby nahrávání.

Je-li nahrávání vypnuto, při každém pokusu o přečtení dalšího čísla ze souboru se systém ohlásí chybovou zprávou

End of file .

Když je nahrávání zapnuto, vykonají se následující akce:

- zazní zvoneček,
- změní se barva okraje obrazovky,
- program vyžaduje nahrání dalšího bloku souboru,
- nahraje další blok textového souboru a otevře ho pro čtení.

Před nahráním musí být magnetofon nastaven na správný soubor, protože FI nahrává podobně jako při použití **LOAD " " CODE**, nekontroluje tedy názvy souborů.

Pokud dojde během nahrávání k chybě, vytiskne se zpráva

Tape loading error .

Kdyby došlo k pokusu nahrát větší soubor, než dovoluje velikost přidělené paměti, ukončí se nahrávání zprávou

Out of memory .

Jestliže chcete číst opakovaně z jednoho souboru, musíte před každým opakováním nastavit čítač na začátek. To provedete příkazem:

RAND USR warm .

V datové části textového souboru je možné používat nejen číselné konstanty, ale i aritmetické výrazy s identifikátory známých proměnných. Pokud se však místo číselné hodnoty výrazu načte nesmyslná posloupnost znaků, vytiskne se chybová zpráva

Nonsense in BASIC .

V datové části mohou být i řetězce znaků. Řetězce se nepíšou do uvozovek. Jestliže je řetězec znaků složen z několika slov (tedy obsahuje mezery), musí být ohraničen hranatými závorkami.

Při čtení ze souboru musíte dbát na to, aby bylo zachováno pořadí a typ

hodnot pro správné načtení do proměnných.

Program FILE IN je vytvořen také ve dvou verzích:

1. *file-in.l*

umístění 31500 až 31986

2. *file-in.h*

umístění 63500 až 63986

Dále použijte hodnoty dle verze:

verze	file-in.l	file-in.h
start cold	31500	63500
start warm	31503	63503
TOP	31955	63955
N.LIN	31957	63957
N.COL	31959	63959
FLAG	31968	63968

Definice umístění v paměti a délky souboru

Provede se stejně jako u programu FO.

Spuštění programu

Provede se také použitím

RANDOMIZE USR cold

1. vyprázdní předem definovaný soubor (vyplní mezerami),
2. nastaví čtení z tohoto souboru,
3. nastaví čítač na začátek.

RANDOMIZE USR warm

2. - 3. jako u cold

Volba nahrávání

Nahrávání souborů můžete zvolit kdykoliv použitím

POKE FLAG, volba

Implicitně: Ne (0)

Příklad použití FI

Napište v editoru (TASWORD, SPECTRAL WRITER) tento text:

```
počet položek { 5 }
položky: {
[leden az brezen] 31+28+pres-
trok+31
[duben az červen] 30+31+30
cervenec 31
srpen 31
[září az prosinec] 30+31+30+31
} konec dat
```

a nahrajte ho na kazetu.

Napište program:

```
10 RANDOMIZE USR 31503:
REM nastavi FI čítač na zač. souboru
20 LET s=0
30 LET PrestRok=0
40 INPUT #3;n:
REM načte první hodnotu
```

```
50 FOR i=1 TO n
60 INPUT #3;p;p:
REM načte řetězec a číslo
70 PRINT p$; TAB 20; p
80 LET s=s+p
90 NEXT i
100 PRINT "soutok";TAB 20;s
```

Nahrajte program FI:

```
CLEAR 29999
LOAD "file-in.l" CODE 31500
RAND USR 31500
```

Nastavte volbu nahrávání:

```
POKE 31968,1
```

Spusťte program v BASICu:

```
RUN
```

Nahrajte soubor (spusťte magnetofon).

Ukázku můžete vyzkoušet také zároveň s TASWORDEM. Postup by byl následující:

- 1) až 4) viz Použití FO,
 - 5) nahrajte program FI,
 - 6) inicializujte FI,
 - 7) napište v TASWORDu vstupní soubor,
 - 8) napište program v BASICu,
 - 9) a spusťte ho
- RUN.

NAPOJENÍ PROGRAMŮ FO A FI NA SYSTÉM

Používání čtení a zápisu z/do textového souboru je umožněno řešením vstupu a výstupu v systému počítače Spectrum.

Systém je vytvořen tak, že komunikace mezi hlavním programem a periférií (implicitně obrazovka, tiskárna a klávesnice) probíhá sériově po jednom znaku (bajtu). Například při vysílání textu se berou postupně jednotlivé znaky (bajty) a předávají se obslužnému programu, který zajišťuje další zpracování znaku. Obdobně to vypadá při načítání znaků.

Právě FO a FI jsou obslužné programy, které zajistí komunikaci mezi hlavním programem a souborem.

Pro vstupy a výstupy je u Spectra možné využívat až 16 proudů (0 až 15). Každý z těchto proudů je možné připojit k některému ze zařízení:

V základním stavu:

K - klávesnice + spodní část obrazovky,

S - horní část obrazovky (pouze výstup),

P - ZX tiskárna (pouze výstup).

s Interface 1:

B - vysílání kódů přes RS 232,

T - vysílání textu přes RS 232,

N - lokální počítačová síť,

M - mdrive.

Implicitně jsou spojeny:

proud 0 na K,

proud 1 na K,

proud 2 na S,

proud 3 na P.

Ostatní proudy jsou uzavřeny.

S proudy jsou také spojeny příkazy BASICu, implicitně:

INPUT - proud 1 (vstup a výstup),

INKEY\$ - proud 1 (vstup),

PRINT - proud 2 (výstup),

LPRINT - proud 3 (výstup).

K propojení proudu s požadavým zařízením dojde použitím

OPEN #proud,"zařízení"

Příkazem OPEN# tedy můžete přepínat i tisk:

OPEN #2,"P"

způsobí, že tisk, který jde normálně na S (obrazovka), vysílá se nyní na P (tiskárna, textový soubor).

OPEN #2,"S"

vrátí tisk zpět na obrazovku.

Stejně tak program, který používá příkaz LPRINT pro tiskárnu, bude po použití

OPEN #3,"S"

tisknout na obrazovku.

BASICové příkazy se také mohou propojit s potřebným proudem

příkaz # proud

např.:

INPUT #3

PRINT #3 (totožné s LPRINT).

Po inicializaci FO a FI (RANDOMIZE USR start) je pro zápis do souboru možné použít příkazy LPRINT a po použití OPEN #2,"P" také příkaz PRINT.

Pro čtení ze souboru je potřeba použít vždy příkaz INPUT #3 bez vložených tisků.

Podle uvedeného popisu je zřejmé, že programy FILE OUT a FILE IN mohou pracovat nezávisle na sobě. Při použití obou programů najednou je výhodné definovat jiný soubor pro FO a jiný soubor pro FI.

```

10 ;
20 ;
30 ; FILE - OUT
40 ;
50 ; program pro vytvoreni
60 ; textovych a datovych souboru
70 ;
80 ; Pavel Kriz (c) 1989
90 ;
100 ;
110 low :EQU 30000
120 high :EQU 64000
130 ;
140 ;
150 ; COND ver?=low
160 *OPENOUT "file-out.1"
170 ;
180 ; ELSE
190 *OPENOUT "file-out.h"
200 ENDC
210 ;
220 ; ORG ver
230 ;
240 ;
250 ;
260 PAUSE:MACRO \P1
270 LD B,\P1
280 P.LBL:HALT
290 DJNZ P.LBL
300 ENDM
310 ;
320 SUB :MACRO \P1,\P2
330 AND A
340 SBC \P1,\P2
350 ENDM
360 ;
370 ;
380 ;
390 ;
400 ;
410 ;
420 ;
430 ;
440 ;
450 ;
460 ;
470 ;
480 ;
490 ;
500 ;
510 ;
520 ;
530 ;
540 ;
550 ;
560 ;
570 ;
580 ;
590 ;
600 ;
610 ;
620 ;
630 ;
640 ;
650 ;
660 ;
670 ;
680 ;
690 ;
700 ;
710 ;
720 ;
730 ;
740 ;
750 ;
760 ;
770 ;
780 ;
790 ;

```

```

800 DEC A
810 SUB B
820 LD HL,LINE
830 CP (HL)
840 RET NC
850 NEXT.PAGE:
860 LD HL,(TOP.PAGE)
870 LD A,(PAGE.LEN)
880 LD C,A
890 LD B,0
900 ADD HL,BC
910 LD (TOP.PAGE),HL
920 LD (T.LIN),HL
930 LD DE,(N.LIN)
940 \SUB HL,DE
950 NC,FULL
960 XOR A
970 LD (LINE),A
980 RET
990 ;
1000 ;
1010 ;
1020 ;
1030 ;
1040 ;
1050 ;
1060 ;
1070 ;
1080 ;
1090 ;
1100 ;
1110 ;
1120 ;
1130 ;
1140 ;
1150 ;
1160 ;
1170 ;
1180 ;
1190 ;
1200 ;
1210 ;
1220 ;
1230 ;
1240 ;
1250 ;
1260 ;
1270 ;
1280 ;
1290 ;
1300 ;
1310 ;
1320 ;
1330 ;
1340 ;
1350 ;
1360 ;
1370 ;
1380 ;
1390 ;
1400 ;
1410 ;
1420 ;
1430 ;
1440 ;
1450 ;
1460 ;
1470 ;
1480 ;
1490 ;
1500 ;
1510 ;
1520 ;
1530 ;
1540 ;
1550 ;
1560 ;
1570 ;
1580 ;
1590 ;
1600 ;
1610 ;
1620 ;
1630 ;
1640 ;
1650 ;
1660 ;

```

```

1670 TAB.2:JR NC,TAB.3
1680 PUSH BC
1690 CALL HL,2
1700 TAB.3:LD (HL),C
1710 RET
1720 ;
1730 ;
1740 BEL :PUSH IX
1750 PUSH HL
1760 LD HL,388
1770 LD DE,105
1780 CALL #385 ;BEEPER
1790 POP IX
1800 \PAUSE 5
1810 RET
1820 ;
1830 FULL :LD HL,FLAG.2
1840 SET FFU,(HL)
1850 DEC HL
1860 BIT FS,(HL)
1870 JP Z,#15E4 ;REPORT.8
1880 CALL BEL
1890 CALL WAIT
1900 CALL FIND
1910 LD (FILE.LEN),BC
1920 LD (ESC.ADR),HL
1930 CALL PE,SAVE
1940 CALL CLEAR.1
1950 LD HL,FLAG.2
1960 RES FFU,(HL)
1970 JP TOF
1980 ;
1990 ;
2000 ;
2010 ;
2020 ;
2030 ;
2040 ;
2050 ;
2060 ;
2070 ;
2080 ;
2090 ;
2100 ;
2110 ;
2120 ;
2130 ;
2140 ;
2150 ;
2160 ;
2170 ;
2180 ;
2190 ;
2200 ;
2210 ;
2220 ;
2230 ;
2240 ;
2250 ;
2260 ;
2270 ;
2280 ;
2290 ;
2300 ;
2310 ;
2320 ;
2330 ;
2340 ;
2350 ;
2360 ;
2370 ;
2380 ;
2390 ;
2400 ;
2410 ;
2420 ;
2430 ;
2440 ;
2450 ;
2460 ;
2470 ;
2480 ;
2490 ;
2500 ;
2510 ;
2520 ;
2530 ;
2540 ;
2550 ;

```

```

2560 LD (IX+10),A
2570 RET
2580 MN.2 :LD A,"0"-1
2590 MN.3 :INC HL,DE
2600 JR NC,MN.3
2610 ADD HL,DE
2620 RET
2630 COMPRIM: LD HL,(ESC.ADR)
2640 INC HL
2650 PUSH HL
2660 LD HL,(TOP)
2670 LD D,H
2680 LD E,L
2690 COM.1: POP BC
2700 SUB HL,BC
2710 RET NC
2720 PUSH BC
2730 LD A,(HL)
2740 LD (DE),A
2750 INC HL
2760 INC DE
2770 CP (HL)
2780 JR NZ,COM.1
2790 LD B,164
2800 COM.2: INC B
2810 JR NZ,COM.3
2820 EX DE,HL
2830 LD (HL),255
2840 EX DE,HL
2850 INC B,165
2860 COM.3: INC HL
2870 CP (HL)
2880 JR Z,COM.2
2890 LD A,B
2900 LD (DE),A
2910 INC DE
2920 LDIR
2930 RET
2940 INIT :CALL CALC
2950 LD HL,(N.LIN)
2960 LD DE,80
2970 SUB HL,DE
2980 JR NC,INIT.1.1
2990 ADD HL,DE
3000 EX DE,HL
3010 INIT.1.1: LD A,E
3020 LD (PAGE.LEN),A
3030 LD (MAX.PL),A
3040 LD A,1
3050 LD (FILE.NUM),A
3060 XOR A
3070 LD (FLAG.2),A
3080 INIT.2: XOR A
3090 LD (L.MAR),A
3100 LD (B.MAR),A
3110 LD (N.COL),A
3120 LD (R.MAR),A
3130 RRCA
3140 AND 63
3150 LD (VT.INC),A
3160 LD A,1
3170 LD (VT.INC),A
3180 LD A,%10001111
3190 LD (FLAG),A
3200 RET
3210 TOF :LD HL,0
3220 LD (T.LIN),HL
3230 LD (TOP.PAGE),HL
3240 LD A,(L.MAR)
3250 LD (COL),A
3260 XOR A
3270 LD (LINE),A
3280 RET
3290 CALC :LD HL,(N.LIN)
3300 LD A,(N.COL)
3310 LD D,0
3320 CALL #30A9 ;MULT
3330 LD (DIM.LEN),HL
3340 RET
3350 RANGE: LD A,(R.MAR)
3360 RANGE.2: LD E,A
3370 DEC E
3380 RANGE.3: LD D,0
3390 RANGE.4: LD A,C
3400 CP D
3410 JR C,RANGE.5
3420 RET
3430 RANGE.5: CP C
3440 RET Z
3450 POP HL
3460 JP ERROR

```

```

3390 ESCAPE: LD DE,ESC.P1
3400 JR ESC.OUT
3410 LD (PARAMS),A
3420 LD C,A
3430 LD HL,TABLE.2
3440 LD B,(HL)
3450 ESC.P1.1: LD HL
3460 INC HL
3470 CP (HL)
3480 JR Z,ESC.P1.2
3490 INC HL
3500 INC HL
3510 DJNZ ESC.P1.1
3520 CALL ESC.END
3530 JP ERROR
3540 ESC.END: LD DE,PRINT
3550 JR #AB0 ;PO.CHANGE
3560 ESC.OUT: LD A,(HL)
3570 LD (T.PARAM),A
3580 INC HL
3590 LD E,(HL)
3600 LD D,(HL)
3610 LD (ESC.ADR),DE
3620 AND A
3630 Z,ESC.JP
3640 DE,ESC.P2
3650 JR ESC.OUT
3660 ESC.P2.1: LD C,A
3670 LD A,(T.PARAM)
3680 LD B,A
3690 LD HL,PARAMS+N.PARAM
3700 DEC HL
3710 DJNZ ESC.P2.1
3720 LD (HL),C
3730 DEC A
3740 LD (T.PARAM),A
3750 RET NZ
3760 ESC.JP: CALL ESC.END
3770 LD HL,(ESC.ADR)
3780 LD A,C
3790 LD (HL),A
3800 ESC.f: AND A
3810 RET Z
3820 LD A,(PARAMS+1)
3830 AND 1
3840 LD B,C
3850 JR NZ,ESC.f1
3860 ESC.f0: PUSH BC
3870 CALL NEXT.COL
3880 POP BC
3890 LD A,(COL)
3900 LD HL,L.MAR
3910 CP (HL)
3920 RET Z
3930 DJNZ ESC.f0
3940 ESC.f1: PUSH BC
3950 CALL NL.2
3960 POP BC
3970 LD A,(LINE)
3980 AND A
3990 RET Z
4000 ESC.f: DJNZ ESC.f1
4010 ESC.e: LD A,(PARAMS+1)
4020 AND 1
4030 JR NZ,ESC.e1
4040 LD E,32
4050 CALL RANGE.3
4060 LD (HT.INC),A
4070 RET
4080 ESC.e1: LD A,(PAGE.LEN)
4090 LD E,A
4100 CALL RANGE.3
4110 LD (VT.INC),A
4120 RET
4130 ESC.C: LD D,1
4140 LD A,(MAX.PL)
4150 LD E,A
4160 CALL RANGE.4
4170 XOR A
4180 LD (B.MAR),A
4190 LD A,(LINE)
4200 AND A
4210 CALL NZ,FF
4220 POP AF
4230 LD (PAGE.LEN),A
4240 ESC.N: LD A,(PAGE.LEN)
4250 CALL RANGE.2
4260 LD (B.MAR),A
4270 LD NL.4
4280 ESC.1: CALL RANGE
4290 LD HL,COL
4300 CP (HL)
4310 JR C,ESC.1.1
4320 LD (HL),A
4330 ESC.1.1: LD (L.MAR),A
4340 RET
4350 ESC.Q: LD A,(L.MAR)
4360 LD D,A
4370 INC D
4380 LD A,(N.COL)
4390 LD E,A
4400 CALL RANGE.4
4410 LD (R.MAR),A
4420 DEC A

```

```

4370 LD HL,COL
4380 CP (HL)
4390 JP C,NEXT.LINE
4400 ESC.*: CALL CLEAR.*
4410 LD A,1
4420 LD (FILE.NUM),A
4430 LD HL,NAME
4440 LD (ESC.ADR),HL
4450 LD A,10
4460 LD (T.PARAM),A
4470 LD DE,ESC.*.1
4480 JP ESC.OUT
4490 ESC.*.1: LD C,A
4500 LD 13
4510 JR Z,ESC.END
4520 LD HL,T.PARAM
4530 LD A,(HL)
4540 AND A
4550 RET Z
4560 DEC (HL)
4570 LD HL,(ESC.ADR)
4580 LD (HL),C
4590 INC HL
4600 LD (ESC.ADR),HL
4610 ESC.n: LD E,100
4620 CALL RANGE.3
4630 LD HL,FILE.NUM
4640 LD (HL),A
4650 ESC.n1: LD A,100
4660 CP (HL)
4670 RET NZ
4680 LD (HL),0
4690 ESC.EOF: LD HL,FLAG.2
4700 SET FFU,(HL)
4710 LD HL,(N.LIN)
4720 TOF.2
4730 ESC.t: LD A,1
4740 JR SET.FLAG
4750 ESC.QU: LD A,2
4760 JR SET.FLAG
4770 ESC.BE: LD A,4
4780 JR SET.FLAG
4790 ESC.p: LD A,8
4800 JR SET.FLAG
4810 ESC.s: LD A,32
4820 JR SET.FLAG
4830 ESC.c: LD A,64
4840 JR SET.FLAG
4850 ESC.w: LD A,128
4860 SET.FLAG: LD HL,FLAG
4870 BIT 0,C
4880 Z,RES.F
4890 LD (HL),A
4900 SET.F: OR (HL)
4910 LD (HL),A
4920 RES.F: AND (HL)
4930 LD (HL),A
4940 TABLE: CARGO
4950 DT1 :MACRO \P1,\P2
4960 DB \P1
4970 DW \P2
4980 ENDM
4990 TABLE.1: DB 16
5000 \DT1 6,COMMA
5010 \DT1 7,BEL
5020 \DT1 9,HT
5030 \DT1 10,NEXT.LINE
5040 \DT1 11,VT
5050 \DT1 12,FF
5060 \DT1 13,NEXT.LINE
5070 \DT1 16,ESC.P1
5080 \DT1 17,ESC.P1
5090 \DT1 18,ESC.P1
5100 \DT1 19,ESC.P1
5110 \DT1 20,ESC.P1
5120 \DT1 21,ESC.P1
5130 \DT1 22,ESC.P1
5140 \DT1 23,ESC.P1
5150 \DT1 27,ESCAPE
5160 ;
5170 DT2 :MACRO \P1,\P2,\P3
5180 DB \P1
5190 DW \P2
5200 \P3
5210 ENDM
5220 TABLE.2: DB 26
5230 \DT2 16,1,ERROR
5240 \DT2 17,1,ERROR
5250 \DT2 18,1,ERROR
5260 \DT2 19,1,ERROR
5270 \DT2 20,1,ERROR
5280 \DT2 21,1,ERROR
5290 \DT2 22,1,AT
5300 \DT2 23,2,TAB
5310 \DT2 *,"0,ESC.*
5320 \DT2 *,"1,ESC.QU
5330 \DT2 *,"0,INIT.2
5340 \DT2 *,"1,ESC.C
5350 \DT2 *,"N,1,ESC.N
5360 \DT2 *,"0,1,ESC.Q
5370 \DT2 *,"1,0,ESC.EOF
5380 \DT2 *,"1,1,ESC.BE
5390 \DT2 *,"1,1,ESC.C
5400 \DT2 *,"2,1,ESC.E
5410 \DT2 *,"1,2,ESC.F
5420 \DT2 *,"1,1,ESC.I
5430 \DT2 *,"1,1,ESC.N
5440 \DT2 *,"1,1,ESC.O
5450 \DT2 *,"1,1,ESC.S
5460 \DT2 *,"1,1,ESC.T

```

```

5350 \DT2 "w",1,ESC.w
5360 ;
5370 ;
5380 VARIABLES:
5390 CARGO
5400 TOP:DW 32000
5410 N.LIN:DW 320
5420 N.COL:DB 64
5430 HEAD:DB 3;TYPE
5440 NAME:DS 10
5450 FILE.LEN:
5460 DW 0
5470 DW 32000 ;TSW.START
5480 FILE.NUM:
5490 DW 0
5500 DIM.LEN:
5510 DW 0
5520 MAX.PL:
5530 DB 0
5540 PAGE.LEN:
5550 DB 0
5560 TOP.PAGE:
5570 DW 0
5580 T.LIN:DW 0
5590 LINE:DB 0
5600 COL:DB 0
5610 L.MAR:DB 0
5620 R.MAR:DB 0
5630 B.MAR:DB 0
5640 HT.INC:
5650 DB 0
5660 VT.INC:
5670 DB 0
5680 FLAG:DB 0
5690 FLAG.2:
5700 DB 0
5710 ESC.ADR:
5720 DW 0
5730 N.PARAM:
5740 EQU 3
5750 PARAMS:
5760 DS N.PARAM
5770 T.PARAM:
5780 DB 0
5790 ;
5800 ;
5810 ;
5820 ;
5830 END :CARGO
5840 ;
5850 ;
5860 PROG.LEN:
5870 EQU
5880 *PRINT "END-START
5890 *PRINT "zacatek",START
5900 *PRINT "konec",END
5910 *PRINT "delka",PROG.LEN
5920 *PRINT "promenne",VARIABLES
5930 ;
5940 ;
5950 *CLOSEOUT

```

```

10 ;
20 ;
30 ;
40 ;
50 ;
60 ;
70 ;
80 ;
90 ;
100 ;
110 ;
120 ;
130 ;
140 ;
150 ;
160 ;
170 ;
180 ;
190 ;
200 ;
210 ;
220 ;
230 ;
240 ;
250 ;
260 ;
270 ;
280 ;
290 ;
300 ;
310 ;
320 ;
330 ;
340 ;
350 ;
360 ;
370 ;
380 ;
390 ;
400 ;
410 ;

```

FILE - IN

program pro cteni z
textovych a datovych souboru

Pavel Kriz (c) 1989

low :EQU 31500
high :EQU 63500

ver :EQU low

COND ver?=low
*OPENOUT "file-in.l"
ELSE
*OPENOUT "file-in.h"
ENDC

ORG ver

FLAG.2
FEOF:EQU 0
FID:EQU 1
FR:EQU 2
FS:EQU 3
FCR:EQU 4

PAUSE:MACRO \P1
LD B,\P1
P.LBL:HALT
DJNZ P.LBL
ENDM

SUB :MACRO \P1,\P2
AND A
SBC \P1,\P2
ENDM

START:CARGO
COLD:CALL CLEAR
WARM:LD A,3
CP #10
JP NC,#15C4 ;REPORT.J
ADD A,A
ADD A,#16
LD L,A
LD H,#5C
LD HL,(HL)
INC HL
LD D,(HL)
LD A,D
OR E
JP Z,#160E ;REPORT.O
INC DE
LD HL,(#3C4F) ;CHANS
ADD HL,DE
LD DE,INPUT
LD (HL),E
INC HL
LD (HL),D

INIT:LD HL,FLAG

```

420 INIT.2:
430 LD HL,(HL),0
440 INC HL
450 LD HL,(HL),0
460 LD HL,(N.LIN)
470 LD A,(N.COL)
480 LD E,A
490 LD D,0
500 CALL #30A9 ;MULT
510 LD (DIM.LEN),HL
520 INIT.3:
530 EX DE,HL
540 LD HL,(TOP)
550 LD (CURS),HL
560 PUSH HL
570 ADD HL,DE
580 LD (EOF),HL
590 POP HL
600 RET
610 CLEAR:CALL INIT.2
620 LD B,D
630 LD C,E
640 LD D,H
650 LD E,L
660 INC DE
670 DEC BC
680 LDIR
690 RET
700 INPUT:LD HL,FLAG.2
710 BIT FE0F,(HL)
720 JP NZ,00F
730 BIT FR,(HL)
740 JP NZ,REP.1
750 BIT FID,(HL)
760 INP.2:CALL Z,DATA.B
770 NEXT.A:
780 CALL GET.A
790 LD HL,FLAG.2
800 CP 165
810 JR NC,REPEAT
820 LD (L.CHAR),A
830 CP " "
840 JR Z,SPACE
850 CP 91
860 JR Z,STR.B
870 CP 93
880 JR Z,STR.E
890 CP 125
900 JR Z,DATA.E
910 SET FCR,(HL)
920 IN.CHAR:
930 PUSH IX
940 LD (HELP),SP
950 LD IX,(HELP)
960 LD E,(IX+8)
970 LD D,(IX+9)
980 LD HL,#F3B
990 \SUB HL,DE
1000 JR NZ,IN.CH.2
1010 LD (IX+8),#4B
1020 IN.CH.2:
1030 POP IX
1040 SCF
1050 SET SV:
1060 RES 1,(IX+1) ;FLAGS
1070 RES 3,(IX+2) ;TV.FLAG
1080 RET
1090 SPACE:BIT FS,(HL)
1100 JR NZ,IN.CHAR
1110 BIT FCR,(HL)
1120 JR NZ,CR
1130 SP.0:CALL SP.1
1140 JR NEXT.A
1150 SP.1:LD HL,(CURS)
1160 DEC HL
1170 SP.2:INC HL
1180 LD A,(HL)
1190 CP " "
1200 JR Z,SP.2
1210 CP 165
1220 JR NC,SP.2
1230 CONTR
1240 STR.B:SET FS,(HL)
1250 BIT FCR,(HL)
1260 JR NZ,CR
1270 SET FCR,(HL)
1280 NEXT.A
1290 STR.E:RES FS,(HL)
1300 JR CR
1310 DATA.E:
1320 RES FID,(HL)
1330 RES FS,(HL)
1340 BIT FCR,(HL)
1350 JP Z,INP.2
1360 CR:RES FCR,(HL)
1370 LD A,13
1380 JR IN.CHAR
1390 REPEAT:
1400 LD (N.REP),A
1410 LD A,(L.CHAR)
1420 CP " "
1430 JR NZ,REP.2
1440 BIT FS,(HL)
1450 JR Z,SP.0
1460 REP.2:LD A,(N.REP)
1470 DEC A
1480 LD (N.REP),A
1490 CP 164
1500 JR NZ,REP.3
1510 REP.3:LD A,(L.CHAR)
1520 LD (L.CHAR),A
1530 IN.CHAR
1540 GET.A:LD HL,(CURS)
1550 LD A,(HL)
1560 INC HL
1570 CONTR:LD (CURS),HL
1580 EX DE,HL

```

```

1300 LD HL,(EOF)
1310 \SUB HL,DE
1320 JR C,00F.0
1330 RET
1340 DATA.B:
1350 LD HL,(EOF)
1360 LD DE,(CURS)
1370 \SUB HL,DE
1380 LD B,H
1390 LD C,L
1400 INC BC
1410 EX DE,HL
1420 LD A,123
1430 CPJR
1440 JP PO,00F.0
1450 LD (CURS),HL
1460 LD HL,FLAG.2
1470 SET FID,(HL)
1480 RET
1490 OOF.0:POP HL
1500 LD HL,FLAG.2
1510 LD DE,FE0F,(HL)
1520 DEC HL
1530 LD A,(HL)
1540 AND A
1550 JR NZ,00F.2
1560 INC A
1570 RET
1580 OOF.2:CALL LD.HEAD
1590 LD LD.FILE
1600 CALL HL,FLAG.2
1610 LD FE0F,(HL)
1620 LD HL,(FILE.LEN)
1630 CALL INIT.3
1640 JP INPUT
1650 LD.HEAD:
1660 PUSH IX
1670 BEL:LD HL,179
1680 LD DE,209
1690 CALL #3B5 ;BEEPER
1700 WAIT:LD A,(#5C4B) ;BORDC
1710 RRCA
1720 RRCA
1730 RRCA
1740 AND 7
1750 LD C,A
1760 XOR A
1770 BIT 2,C
1780 JR NZ,WAIT.2
1790 LD A,7
1800 WAIT.2:
1810 OUT (254),A
1820 \PAUSE 5
1830 PUSH BC
1840 WAIT.3:
1850 LD B,#9C
1860 CALL #5E3 ;LD.EDGE.2
1870 JR C,WAIT.4
1880 JR Z,WAIT.3
1890 WAIT.4:
1900 POP BC
1910 LD A,C
1920 OUT (254),A
1930 JP NC,#552 ;REPORT.D
1940 LD IX,HEAD
1950 LD DE,17
1960 XOR A
1970 SCF
1980 CALL #556 ;LD.BYTES
1990 JR NC,BEL
2000 LD A,(HEAD)
2010 CP 3
2020 JR NZ,BEL
2030 LD DE,(HEAD+11)
2040 LD HL,(DIM.LEN)
2050 \SUB HL,DE
2060 POP IX
2070 RET NC
2080 CALL SET.SV
2090 RST 8
2100 DB 3 ;REPORT.4
2110 LD.FILE:
2120 PUSH IX
2130 LD IX,(TOP)
2140 LD A,255
2150 SCF
2160 CALL #556 ;LD.BYTES
2170 POP IX
2180 JP NC,#806 ;REPORT.R
2190 RET
2200 VARIABLES:
2210 TOP:DW 32000
2220 N.LIN:DW 320
2230 N.COL:DB 64
2240 DIM.LEN:
2250 DW 0
2260 EOF:DW 0
2270 CURS:DW 0
2280 L.CHAR:
2290 DB 0
2300 N.REP:DB 0
2310 FLAG:DB 0
2320 FLAG.2:
2330 DB 0
2340 HEAD:DS 17
2350 HELP:EQU HEAD
2360 FILE.LEN:
2370 EQU HEAD+11
2380 ;
2390 ;
2400 END :CARGO
2410 PROG.LEN:
2420 EQU
2430 *PRINT "zacatek",START
2440 *PRINT "konec",END
2450 *PRINT "delka",PROG.LEN
2460 *PRINT "promenne",VARIABLES
2470 ;
2480 ;
2490 *CLOSEOUT

```

SIMULÁTOR

JEDNOČIPOVÝCH MIKROPOČÍTAČŮ

Ing. Roman Čech, Kalnara 903, 721 00 Ostrava - Svinov

Světovým standardem jednočipových mikropočítačů se stala řada MCS-48, kterou vyvinula v roce 1976 firma Intel. V letech 1984 - 1985 byly zahájeny dodávky jednočipových mikropočítačů řady 8048 na československý trh. Jedná se o osmibitové mikropočítače, které v jedlém pouzdře se čtyřiceti vývody obsahují základní jednotku, oddělené adresovatelné paměti programu a dat, vstupní/výstupní obvody, čítač/časovač, logiku přerušení aj.

Mikropočítačovou řadu MCS-48 tvoří obvody 8035, 8048, 8748 a jejích modifikace (viz Tab. 1). Jednotlivé obvody se od sebe liší druhem a kapacitou paměti, technologií výroby, pracovním kmitočtem a možností samostatného napájení vnitřní paměti registrů (odpojení zbytku obvodu v neaktivní době).

1.1. Základní vlastnosti mikro-počítačů řady MCS-48

Popis uvedený v této kapitole odpovídá mikropočítači 8048.

Z hlediska programátora má mikro-počítač 8048 tyto základní vlastnosti:

- paměť programu typu ROM o velikosti 1024 osmibitových slabik s možností rozšíření až na velikost 4096 osmibitových slabik,
- paměť dat typu RAM o velikosti 64 osmibitových slabik, která obsahuje dvě sady registrů a zásobník, paměť dat lze doplnit o vnější paměť s kapacitou až 256 osmibitových slabik,
- stavové slovo programu (PSW), které obsahuje stavové bity, příznaky a ukazatel zásobníku,
- střadač (A) sloužící jako cílový či zdrojový registr pro aritmeticko-logické operace nebo pro čtení a zápis dat,
- dvanáctibitový čítač instrukcí,
- programovatelný čítač/časovač,
- možnost vnějšího přerušení a přerušení při přetečení čítače/časovače,
- vstupní/výstupní porty, rozšiřitelné použitím obvodu expanderu 8243.

1.1.1. Struktura paměti programu

Integrovaný obvod jednočipového mikropočítače 8048 obsahuje 1024 osmibitových slabik pevné paměti programu, kterou lze rozšířit použitím pří-
davných pamětí až na 4096 osmibito-

Typ	paměť programu	paměť dat	Kanály V/V	Vnější přerušení	Možnost rozšíření	Převodník A/D
8048	1kB/ROM	64B	3x8	ANO	ANO	NE
8035	-	64B	3x8	ANO	ANO	NE
8748	1kB/EPROM	64B	3x8	ANO	ANO	NE
8049	2kB/ROM	128B	3x8	ANO	ANO	NE
8039	-	128B	3x8	ANO	ANO	NE
8749	2kB/EPROM	128B	3x8	ANO	ANO	NE
8041	1kB/ROM	64B	4x8	NE	NE	NE
8741	1kB/EPROM	64B	3x8	NE	NE	NE
8021	1kB/ROM	64B	2x8 1x4	NE	NE	NE
8022	2kB/ROM	64B	3x8	NE	NE	

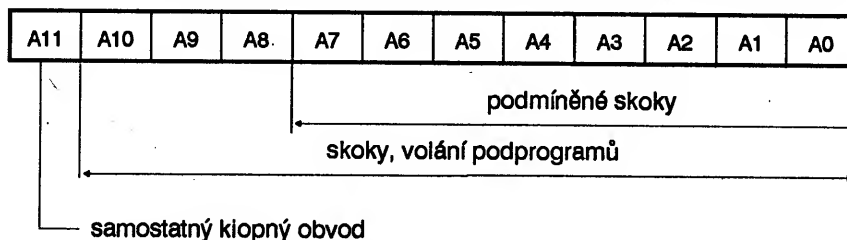
Tab. 1. Modifikace mikropočítačů řady MCS-48

vých slabik. Toto rozšíření sníží počet vstupů a výstupů.

K adresování programové paměti slouží dvanáctibitový čítač (obr. 1), jehož nejvyšší bit (A11) je samostatný klopný obvod, který vybírá ze dvou polovin (bank) paměti programu označených názvem MEMORY BANK 0 (A11=0) a MEMORY BANK 1 (A11=1). Klopný obvod A11 se nastavuje pro-

gramově instrukcemi SEL MB0 a SEL MB1. K vlastnímu nastavení však dojde až při vyvolání instrukce CALL nebo JMP. Při obsluze přerušení je klopný obvod A11 samočinně udržován na 0. Do původního stavu se vrací při příkazu návratu.

S adresováním také souvisí rozdělení paměti na stránky po 256 osmibitových slabikách. U podmíněných



Obr. 1. Čítač programu

skoků je totiž adresování přímé s adresou v druhé slabice příkazu. Těchto 8 bitů se umístí v čítači instrukcí jako bity A0 až A7. Podmíněné skoky lze tak programovat pouze v rozsahu jedné stránky.

V paměti programu jsou tři adresy s pevně přiřazeným významem:

adresa 0H - začátek uživatelského programu,

adresa 3H - začátek obslužného programu vnějšího přerušení,

adresa 7H - začátek obslužného programu přerušení vyvolaného přetečením čítače/časovače.

V praxi se vzhledem k vlastnostem instrukčního souboru doporučuje využívat jednotlivé stránky programové paměti takto:

0. stránka - obslužné podprogramy přerušení,

1., 2. stránka - uživatelský program,

3. stránka - tabulka konstant (ze třetí stránky je možno číst data a ukládat je do střadače instrukcí MOV P3 A, @A).

Organizace paměti programu je na obr. 3.

1.1.2. Struktura paměti dat

Kromě programové paměti obsahuje obvod mikropočítače 8048 datovou paměť o velikosti 64 osmibitových slabik. Její velikost lze rozšířit vnější pamětí o velikosti až 256 osmibitových slabik, které jsou přístupné pouze instrukcemi MOVX @A, A a MOVX A, @A.

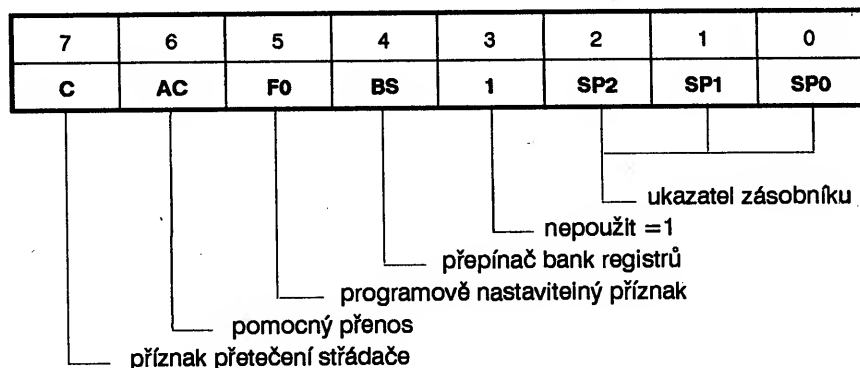
Vnitřní datová paměť se skládá ze dvou skupin pracovních registrů, osmúrovňového zásobníku typu LIFO a vlastní paměti. Organizace datové paměti je na obr. 4.

Každá skupina registrů (banka registrů), jejichž výběr se provádí instrukcemi SEL RB0 a SEL RB1, obsahuje osm pracovních registrů. První banku registrů je vhodné rezervovat pro podprogramy obsluhující přerušení. Registry, které nejsou používány, můžeme využít jako normální slabiky datové paměti.

Pro zásobník je vyhrazeno šestnáct paměťových míst. Při vyvolání podprogramu nebo rutiny přerušení jsou programový čítač (12 bitů) a bity 4 až 7 stavového slova uloženy do dvou slabik zásobníku. Obrácenou činnost vyvolá instrukce návratu z podprogramu RETR. Instrukce RET obnoví pouze programový čítač. Adresace zásobníku je provedena bity 0 až 2 stavového slova.

1.1.3. Stavové slovo programu PSW a informace pro řízení programu

Stavové slovo programu PSW obsahuje 8 bitů. Jejich uspořádání a význam je na obr. 2.



Obr. 2. Stavové slovo programu PSW

Kromě přenosu C a programově nastavitelného příznaku F0, obsahuje mikropočítač 8048 několik dalších vývodů, příznaků a podmínkových bitů,

kteří lze použít k řízení programu. Jsou to:

F1 - programově nastavitelný příznak,

T0 - testovatelný vývod 0,

T1 - testovatelný vývod 1,

TF - příznak přetečení čítače/časovače,

I - vstupní vývod přerušení.

Stav všech těchto jednobitových informací lze testovat podmíněnými skokovými instrukcemi.

1.1.4. Registr časovače a čítače událostí

Registr časovače a čítače je osmibitový registr, který ve funkci časovače zvyšuje svůj stav vždy po 32 instrukčních cyklech. Pracuje-li registr jako čítač událostí, je jeho stav zvyšován signály přivedenými na vývod T1. Hodnotu čítače/časovače a jeho funkci lze nastavit programem. Programově lze také povolit, případně zakázat čtení.

1.1.5. Přerušení

Mikropočítač 8048 rozeznává dva druhy přerušení:

- vnější,
- vnitřní.

Vnější přerušení předá řízení programu na adresu 3H programové paměti. Vnitřní přerušení je vyvoláno přetečením čítače/časovače a předá řízení programu na adresu 7H programové paměti. Obě přerušení lze programově povolovat a zakazovat. Vyskytnou-li se současně, jsou obě rozpoznány, vnější přerušení má vyšší prioritu.

1.1.6. Vstupní a výstupní obvody

Ze 40-ti vývodů mikropočítače 8048 může být 27 použito pro vstup, výstup nebo obojí v závislosti na celkové konfiguraci. Celkový počet vstupních a výstupních linek lze použitím obvodu expanderu 8243 zvýšit na 43 linek, které jsou organizovány takto:

port BUS - osmibitový statický vstup nebo výstup ovládaný signály RD a WD. Po připojení vnější paměti

a)

4095	7. stránka	
3840		
2303	0. stránka	memory bank 1
2048		
2047	7. stránka	memory bank 0
1792		
1791	6. stránka	
1536		
1535	5. stránka	
1280		
1279	4. stránka	vnější
1024		
1023	3. stránka	vnitřní
768		
767	2. stránka	
512		
511	1. stránka	
256		
255	0. stránka	
0		

b)

7	začátek obslužného podprogramu přerušení od čítače/časovače
3	začátek obslužného podprogramu vnějšího přerušení
0	začátek uživatelského programu

Obr. 3. Paměť programu

a) stránkování paměti, b) 0. stránka

63	bez zvláštního určení	
62		
61		
60		
59		
35		
34		
33		
32		
31	R7	1. banka registrů
30	R6	
29	R5	
25	R1	zásobník 8 x 2 bajtů
24	R0	
23		
22		
		0. banka registrů
9		
8		
7	R7'	
6	R6'	
5	R5'	
4	R4'	
3	R3'	
2	R2'	
1	R1'	
0	R0'	

adresa

Obr. 4. Paměť dat

programu přenáší osm nižších bitů programového čítače a přivádí vyvolanou instrukci do mikropočítače. Při spolupráci mikropočítače s vnější datovou pamětí předává adresu i data,

port 1 - osmibitový kvazidvousměrný vstup nebo výstup,

port 2 - osmibitový kvazidvousměrný vstup nebo výstup. Nižší čtyři bity slouží pro připojení vnější paměti k přenosu čtyř nejvyšších bitů programového čítače do paměti. Je-li použit expander 8243, přenáší se jimi kód činnosti a data.

T0 - jednobitový testovatelný vstup, na který lze vyvést hodinové pulsy systému pro vnější použití,

T1 - jednobitový testovatelný vstup, který slouží jako vstup čítače událostí,

INT - jednobitový testovatelný vstup vnějšího přerušení,

port 4-7 - čtyřbitové obousměrné vstupy nebo výstupy obvodu expanderu 8243.

S daty na všech portech je možno provádět operace logického součinu a logického součinnu.

1.2. Instrukční soubor

Instrukční soubor jednočipového mikropočítače 8048 je upraven tak, aby se jím daly snadno realizovat programy, týkající se samočinného řízení objektů bez nároků na příliš velký rozsah paměti programu. Lze jej rozdělit do čtyř hlavních funkčních skupin. Jsou to instrukce pro přesun dat, manipulaci s daty, přenos řízení a nastavení parametrů programu. Všechny instrukce mají délku 1 nebo 2 bajty a doba potřebná k jejich provedení při kmitočtu vnějšího oscilátoru 6 MHz je podle počtu strojových cyklů (jeden nebo dva) 2,5 nebo 5 μ s.

Instrukce pro přesun dat slouží k přesunu dat z jednotlivých částí paměti dat do střadače a naopak, k načítání dat z paměti programu do střadače a k přesunu dat mezi porty a střadačem.

Instrukce pro manipulaci s daty zahrnují instrukce pro matematickologické operace s daty ve střadači, registrech a portech, a instrukce pro posuv dat ve střadači.

Mezi instrukce pro předávání řízení programu patří instrukce volání podprogramů a skokové instrukce.

Parametry programu se rozumí nastavení podmínkových bitů, povolení nebo zakázání přerušení, nastavení funkce čítače/časovače, výběr banky registrů a banky paměti.

Popis instrukcí a odlišností instrukčních souborů pro jednotlivé typy mikropočítačů je uveden v literatuře [4].

1.3. Modifikace řady MCS-48 a řady MCS-51

K mikropočítačům základní řady byly postupně vyvinuty další modifikace, které se vyznačují zjednodušením nebo rozšířením možností původní řady.

Mikropočítač 8021 má vlastnosti, které jsou podmnožinou vlastností mikropočítače 8048. Základním rozdílem je zapouzdření obvodu (pouze 28 vývodů) a zrušení portu BUS. To znamená, že je možno používat pouze vestavěné paměti programu a dat. Výběr banky registrů není u mikropočítače 8021 možný.

Mikropočítač 8022 s vnitřní pamětí typu ROM o kapacitě 2048 osmibitových slábk obsahuje navíc analogově-číslíkový převodník. Dva analogové vstupy jsou přepínatelné programově.

Mikropočítače 8041 a 8741 zahrnují v sobě rozhraní a protokoly pro sběrnice nadřazených mikropočítačů (8048, 8080, 8085) a mohou tak sloužit jako programovatelný inteligentní periferní procesor uvnitř větších mikropočítačových systémů.

Mikropočítače řady MCS-51 (8031, 8051, 8751) byly vyvinuty pro náročnější aplikace. Mají větší možnost

přímého adresování vnější paměti programu i dat (2 x 64 KB), větší počet registrů (4 x 8), dva šestnáctibitové čítače/časovače, 128 jednobitových příznaků ovládaných programem. Dále jsou vybaveny programovatelným sériovým kanálem a Booleovským procesorem pro řídicí aplikace.

1.4. Aplikace mikropočítačů řady MCS-48

Uvedené vlastnosti mikropočítačů řady MCS-48 umožňují nasazení mikroelektroniky i v aplikacích, kde osmibitové mikroprocesorové systémy (např. s 8080) vycházejí nákladné.

Nasazení jednočipového mikropočítače řady 8048 je nejvýhodnější v zapojeních, kde se vystačí se samotným mikropočítačem nebo s minimem podpůrných obvodů. Jedná se především o aplikace v řídicích nebo aritmetických jednotkách strojů a přístrojů vyráběných ve velkých sériích, kdy se použije mikropočítač s pamětí programu typu ROM, jejíž obsah je dán již ve výrobě. Toto nasazení je však efektivní až při několika tisíci sériích. Pro jednorázové, laboratorní, vývojové nebo malosériové aplikace je výhodnější použít mikropočítač s pamětí typu EPROM.

2. Vývoj programů pro mikropočítač řady 8048

Vlastnosti instrukčního souboru mikropočítačů řady 8048 neumožňují použít žádný z vyšších programovacích jazyků. K tvorbě programového vybavení je tedy možno použít pouze strojově orientované jazyky. Hlavní problémy při vývoji aplikací s těmito obvody vyplývají z nedostupnosti jejich vnitřní struktury a rozdělení paměti do samostatných skupin.

Z výše uvedených důvodů se při práci s mikropočítači řady 8048 používají vývojové pomůcky, u kterých je paměť programu sestavena z obvodů typu RAM.

Efektivnějšími vývojovými prostředky jsou vývojové systémy s emulátory. Pro přípravné práce na vývoji programového vybavení slouží křížové prostředky provozované na hostitelských počítačích.

2.1. Simulátor

Simulátor je program, který umožňuje uživateli křížového programového vybavení simulovat na hostitelském počítači činnost mikropočítače řady 8048. Pomocí tohoto programu lze tedy ověřit logickou správnost vyvíjeného programu zapsaného ve strojovém kódu. K úplnému ověření simulátor použít nelze, protože neposkytuje informace o hardwarových signálech mezi

jednotlivými prvky mikropočítačového systému.

Programem lze vypisovat a modifikovat obsahy registrů, střadače, stavového slova, čítače/časovače, vstupních a výstupních linek a dalších částí mikropočítače. Ladící činnost lze provádět po jednotlivých instrukcích nebo ucelených blocích. Protože ladění neprobíhá v reálném čase, jsou součástí simulátoru hodiny, které sledují dobu potřebnou k provedení programu.

3. Návrh koncepce programového simulátoru

Návrh koncepce programu vychází z těchto požadavků:

- tvorba vyvíjeného programu pro mikropočítač 8048 na úrovni strojového kódu,
- kontrola vyvíjeného programu zpětným překladačem,
- rychlá simulace všech instrukcí mikropočítače 8048,
- přehlednost výsledků simulace,
- modelování stavu simulovaného mikropočítače,
- možnost uložení a zpětného nahrání odladených programů,
- snadná převoditelnost programu na jiné osobní počítače s mikroprocesorem řady 80 nebo Z80,
- malá náročnost na kapacitu paměti.

Pro jejich splnění je program rozdělen na několik samostatných bloků. Simulaci instrukcí a zpětný překlad provádějí bloky překladače a interpretu, které jsou sestaveny ve strojovém kódu. Ovládání těchto bloků, tvorbu vyvíjeného programu, přehlednost výsledků simulace, modelování stavu a archivaci programů provádí blok obsluhy. Tento blok tedy slouží ke komunikaci s uživatelem a využívá podprogramy pro vstup z klávesnice a výstup na obrazovku (monitor) a pro práci s vnější pamětí (magnetofonem). Programování těchto podprogramů na úrovni strojového kódu a při zachování požadavku na převoditelnost programu na jiné osobní počítače s mikroprocesorem 8080 nebo Z80 je značně problematické. Výsledný program by vyžadoval při převodu na jiný typ počítače od uživatele znalost programování ve strojovém kódu a znalost monitoru hostitelského počítače. Z těchto důvodů je vhodnější naprogramovat blok obsluhy ve vyšším programovacím jazyce. Nejvhodnějším jazykem pro programování bloku obsluhy je BASIC, kterým jsou osobní počítače standardně vybaveny.

Takovéto rozdělení je nutné z důvodu nestandardnosti osobních počítačů. V opačném případě by bylo výhodnější celý program sestavit ve strojovém kódu. Výsledkem by byla několikrát násobně větší rychlost programu.

název	(B)	význam
<i>definice systému</i>		
ZTT	2	adresa začátku kódové tabulky pro tisk překladu disassemblerem
ZPP	2	adresa začátku programové paměti
ZRAM	2	adresa začátku datové paměti
ZRAMP	2	adresa začátku přídatné datové paměti
ZTP	2	adresa začátku tabulky skoků pro simulaci instrukcí
TZ	2	adresa začátku tiskové zóny
<i>vstupy a vnitřní stav mikropočítače</i>		
T0	1	stav vývodu T0
T1	1	stav vývodu T1
INT	1	stav vstupu vnějšího přerušení
PORT0	1	obsah portu BUS
PORT1	1	obsah portu 1
PORT2	1	obsah portu 2
PORT4	1	obsah portu 4 (8243)
PORT5	1	obsah portu 5 (8243)
PORT6	1	obsah portu 6 (8243)
PORT7	1	obsah portu 7 (8243)
AKU	1	obsah střadače (akumulátoru)
PSW	1	obsah stavového slova
TIME	1	obsah registru čítače/časovače
PC48	2	obsah programového čítače
<i>pomocné proměnné</i>		
KONADR	2	koncová adresa bloku pro rychlou simulaci
PRERUS	1	viz Tab. 4.
BYTE	1	délka přeložené instrukce
STRCYK	1	pomocné počítadlo strojových cyklů
HODINY	3	průběžné počítadlo strojových cyklů

Tab. 3. Systémové proměnné

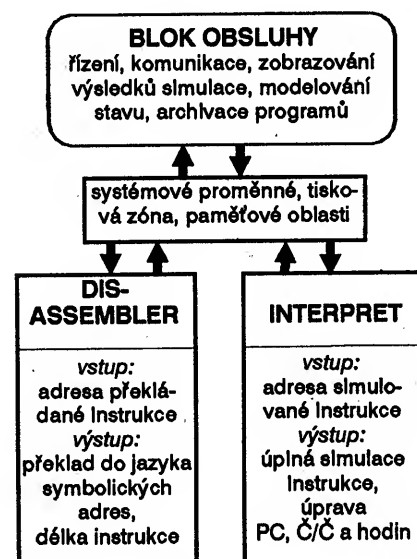
0. bit	obsluha přerušení od čítače/časovače (0 - povolena, 1 - maskována)
1. bit	funkce čítače/časovače (0 - časovač, 1 - čítač)
2. bit	čítání (0 - start, 1 - stop)
3. bit	nastavení banky paměti
4. bit	1 - vývod T0 použit jako výstup hodin
5. bit	příznak přetečení čítače/časovače
6. bit	testovatelný příznak F1
7. bit	obsluha vnějšího přerušení (0 - povolena, 1 - maskována)

Tab. 4. Význam jednotlivých bitů systémové proměnné PRERUS

S ohledem na rychlost je v případě možnosti vhodné použít místo jazyka BASIC, který pracuje jako interpret, jazyk PASCAL nebo FORTRAN. Tyto jazyky pracují jako kompilátory a jsou tedy rychlejší. Podmínkou pro jejich použití je možnost volání strojových podprogramů. Problémy by také mohlo činit ukládání odladených programů.

Na začátku bloku strojového kódu je uložena tabulka proměnných, které slouží k přenášení údajů mezi programem ve strojovém kódu a obslužným programem a k vyhrazení prostorů programové a datové paměti simulovaného mikropočítače. Seznam proměnných, jejich název, délka v bajtech a význam je uveden v Tab. 3.

Struktura navrhovaného programovaného simulátoru je znázorněna na obr. 5.



Obr. 5. Struktura PS-48
Blok obsluhy je v BASICu, disassembler a interpret ve strojovém kódu

4. Realizace programu

Program byl pod názvem PS-48 (programový simulátor mikropočítače 8048) sestaven a odladěn na osobním počítači Sinclair ZX Spectrum. V souladu s dříve uvedenými požadavky jej lze převést na osobní počítače pracující s mikroprocesory 8080 nebo Z80. Jsou to počítače IQ 151, PMD 85, ONDRA, DIDAKTIK ALFA, SHARP MZ800 a další.

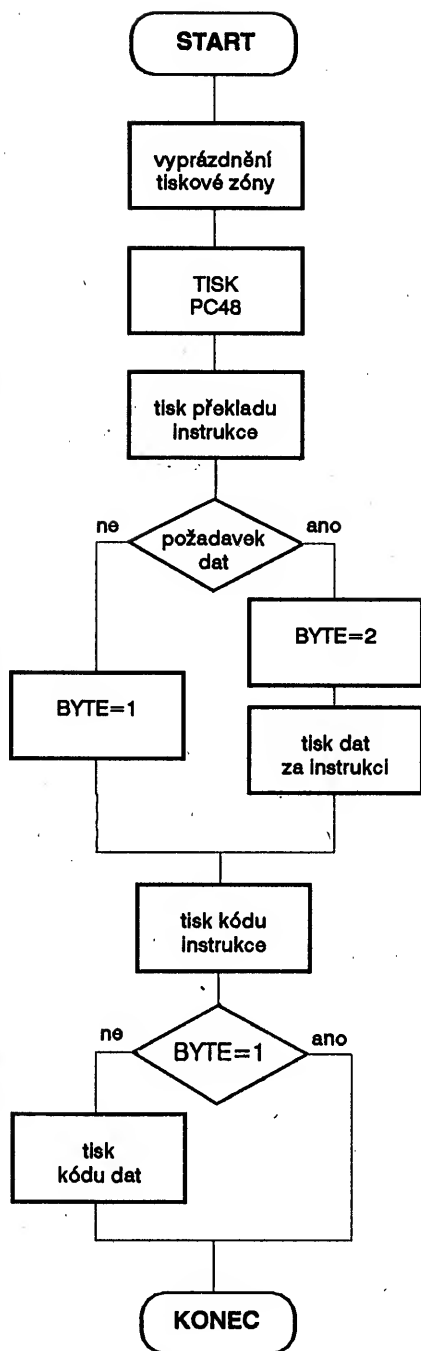
Vzhledem k různému umístění operační paměti u jednotlivých počítačů bude v některých případech nutno provést předadresování bloků napsaných ve strojovém kódu. Pro počítače IQ 151 a PMD 85 lze použít překlad ze str.00.

Při přepisu bloku obsluhy je třeba přihlídnout k použité verzi jazyka BASIC. U počítače PMD 85 není pro blok obsluhy dostatek volné paměti. Z tohoto důvodu je nutno blok obsluhy omezit o některé podprogramy nebo jej vytvořit ve strojovém kódu.

4.1. Blok disassembleru

Tento blok programu je napsán ve strojovém jazyce. Jeho úkolem je zajistit překlad instrukce, která je uložena v programové paměti simulovaného procesoru na adrese uložené v proměnné PC48. Překlad je veden ze strojového kódu do jazyka symbolických adres. Výsledek bude uložen ve formě ASCII znaků do tiskové zóny o délce 20 bajtů. Délka přeložené instrukce (jednobajtová, dvoubajtová) se uloží do proměnné BYTE. Program probíhá podle vývojového diagramu na obr. 6. V první části dojde k vyprázdnění tiskové zóny. Následuje tisk hodnoty proměnné PC48, jejíž obsah je převeden z binárního tvaru do hexadecimálního vyjádření v ASCII znacích. Tisk překladu instrukce se provádí vyhledáním příslušné části v tabulce tisků podle kódu instrukce, a jejím přenesením do tiskové zóny. Adresa začátku tabulky tisků je uložena v proměnné ZTT. Je-li nutno změnit umístění tabulky tisků v paměti (při aplikaci programu na počítačích s různou organizací operační paměti), lze to provést tak, že se tabulka přemístí na požadované místo a do proměnné ZTT se uloží adresa nového začátku tabulky. Překlady jsou v tabulce uloženy pro všechny kódy (od 0 do 255) instrukčního souboru v konstantní délce devíti bajtů. Kódování je provedeno tak, že mezery jsou vypuštěny a kód prvního znaku následujícího po mezeře je uložen se sedmým bitem rovným jedné. Požadavek dat je signalizován znakem #. Pro kódy, jejichž význam není definován, je v tabulce uložen překlad NEDEF.

*Příklad zakódování instrukce
MOV A, data.*



Obr. 6. Vývojový diagram disassembleru

Na příslušném místě, odpovídajícím kódu instrukce bude v tabulce tisků uloženo těchto devět hodnot:

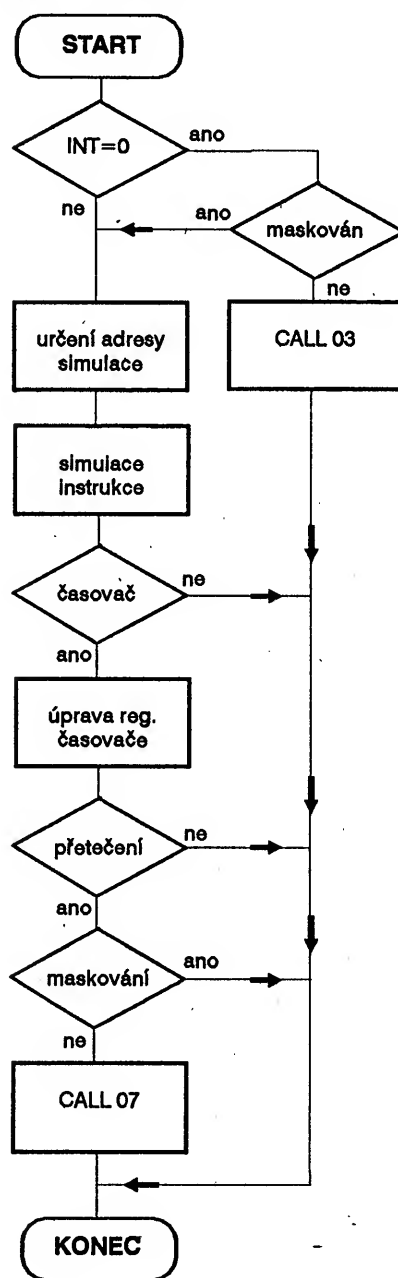
- | | | | |
|----|---|----|--|
| 1. | M | -> | 4D |
| 2. | O | -> | 4F |
| 3. | V | -> | 56 |
| | | -> | ignorování mezery |
| 4. | A | -> | C1 1. znak po mezeře |
| 5. | , | -> | 2C |
| 6. | # | -> | 23 indikace požadavku dat |
| 7. | | -> | 20 doplnění mezerami na konstantní délku |
| 8. | | -> | 20 |
| 9. | | -> | 20 |

V závěru se do tiskové zóny vytiskne kód instrukce a případná data.

4.2. Blok interpretu

Blok Interpretu je napsán ve strojovém kódu a ve spolupráci s blokem disassembleru provádí vlastní simulaci jednotlivých instrukcí mikroprocesoru řady 8048. Jeho činnost je dána vývojovým diagramem na obr. 7.

Realizuje oba druhy přerušení, nastavuje PC48, upravuje hodnoty časovače, zásobníku, registrů, stavového slova atd. Hodnoty registrů a zásobníku jsou uloženy v oblasti datové paměti simulátoru, jejíž začátek je uložen v proměnné ZRAM. Hodnoty akumulátoru, stavového slova, čítače/časovače, portů, programového čítače, vstupů a výstupů jsou uloženy v proměnných podle Tab. 3.



Obr. 7. Vývojový diagram interpretu

Paměť programu simulovaného mikropočítače je vyhrazena oblast velikosti 4096 bajtů, paměť dat oblast velikosti 64 bajtů a externí paměť dat oblast velikosti 256 bajtů. Adresy začátků těchto oblastí v operační paměti počítače jsou uloženy v proměnných ZPP, ZRAM a ZRAMP.

Interpret simuluje instrukci, jejíž kód leží v paměti programu na adrese uložené v proměnné PC48. Adresa v operační paměti počítače je pak dána součtem proměnných PC48 a ZPP.

Samotná simulace jednotlivých instrukcí probíhá tak, že z tabulky podprogramů, ve které jsou uloženy dvou-bajtové adresy začátků podprogramů pro jednotlivé kódy od 0 do 255, se vybere odpovídající adresa a program na ní skočí. Adresa začátku tabulky podprogramů je uložena v proměnné ZTP. Umístění tabulky podprogramů v operační paměti počítače je možno měnit stejně jako v případě tabulky tisků.

Podprogramy simulují činnost odpovídající instrukce. Výsledky simulace se projeví v paměti dat nebo v proměnných, které představují vnitřní stav simulovaného mikropočítače, případně stav jeho vývodů.

Na konci každého podprogramu se upraví proměnná PC48 tak, aby obsahovala adresu další instrukce. V případě skoků a volání podprogramu je hodnota proměnné PC48 dána instrukcí a jejími daty. V ostatních případech se obsah proměnné PC48 zvětší o hodnotu proměnné BYTE.

Proměnné STRCYK a HODINY jsou zvětšovány o počet strojových cyklů, který simulovaná instrukce potřebuje ke své činnosti. Proměnná STRCYK slouží k nastavování časovače. Proměnnou HODINY lze využít k měření času potřebného k provedení daného úseku programu.

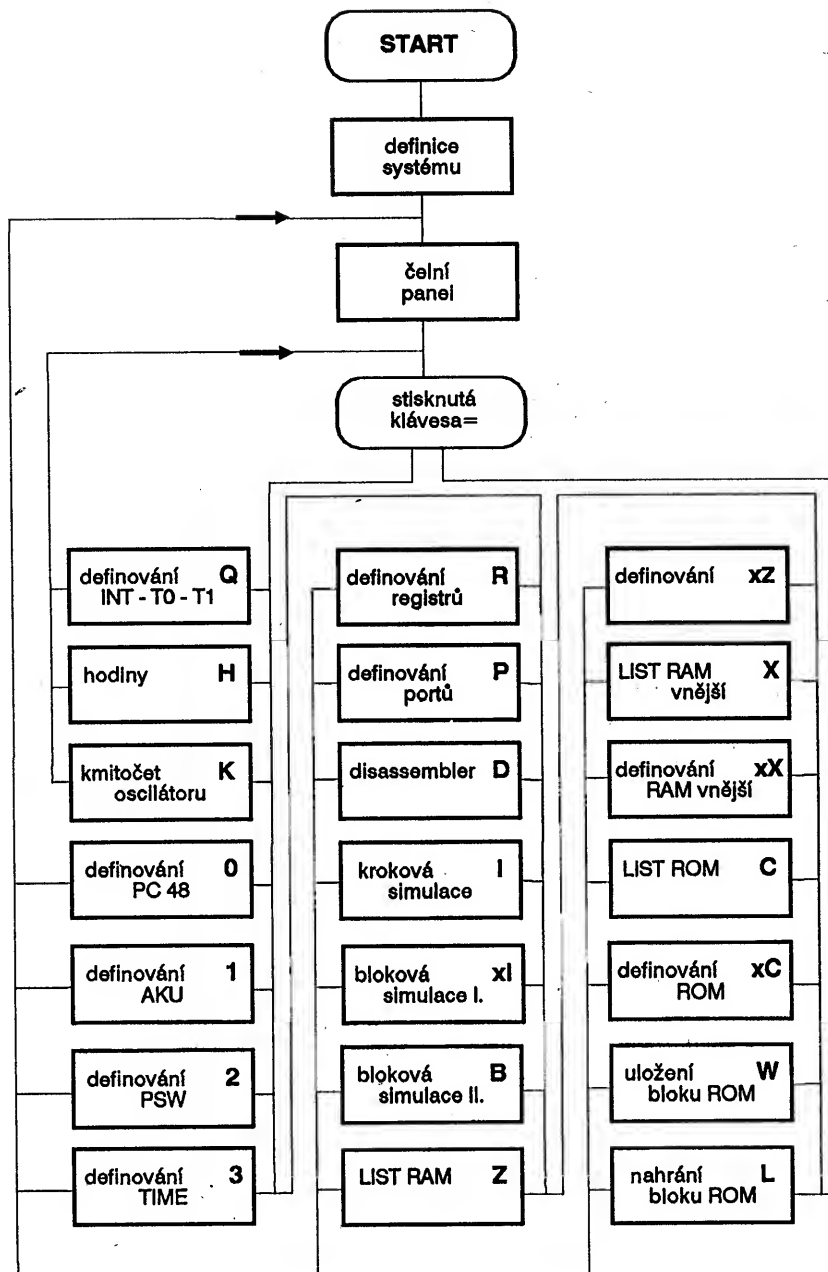
Součástí interpretu je podprogram pro simulaci ocladěných bloků. Spouští se samostatně a vyžaduje v proměnné KONADR adresu instrukce, na které se simulace zastaví.

4.3. Blok obsluhy

Blok obsluhy je napsán v jazyce BASIC. Jeho úkolem je zajistit komunikaci mezi uživatelem a bloky interpretu a disassembleru. Jeho činnost je dána vývojovým diagramem na obr. 8.

Po nahrání a spuštění programu zobrazí blok obsluhy čelní panel, obsahující hodnoty všech registrů, portů, akumulátoru, stavového slova a hodnotu časovače/čítače. V horní části obrazovky se vypisuje hodnota programového čítače a překlad instrukce, která se nachází na adrese dané jeho obsahem.

Uživatelé jsou k dispozici podprogramy pro tvorbu, kontrolu a simulaci vyvíjeného programu, modelování stavu simulovaného mikropočítače a ar-



Obr. 8. Vývojový diagram bloku obsluhy

chivací programů. Podprogramy se vyvolají stiskem příslušné klávesy.

4.3.1. Tvorba vyvíjeného programu

PS-48 není vybaven assemblerem a umožňuje vytvářet programy pouze na úrovni strojového kódu. Podprogram, kterým se ukládají hodnoty do paměti programu, se vyvolá současným stiskem kláves „symbol shift“ a „C“. Po zadání adresy, od které budeme program ukládat, se vypíše obsah buňky paměti programu na zadané adrese. Tento obsah můžeme změnit zadáním nové hodnoty, která se uloží na danou adresu. Klávesou „Enter“ způsobíme vypisování obsahu následujících buňek paměti programu. Návrat do hlavního programu se provede stiskem mezerníku.

K prohlížení paměti programu je určen podprogram, který se vyvolává klávesou „C“. Zadává se adresa, od které se má provést výpis. Každý řádek obsahuje adresu programové paměti a obsah osmi buněk od této adresy. Po vypisování dvaceti řádků se výpis zastaví. Mezerník způsobí návrat do hlavního programu. Ostatní klávesy způsobí pokračování výpisu.

Ke kontrole programu uloženého do paměti je určen podprogram DISASSEMBLER, který se vyvolá klávesou „D“. Podprogram převede obsah programové paměti od adresy v programovém čítači do jazyka symbolických adres. Stisk mezerníku vyvolá návrat do hlavního programu. Stisk jiné klávesy způsobí další překlad. Délka přeloženého úseku je jedenáct instrukcí.

kláv.	význam
xC	definování paměti programu
C	listování paměti programu
xZ	definování paměti dat
Z	listování paměti dat
xx	definování vnější paměti dat
X	listování vnější paměti dat
D	disassembler
R	definování registrů
P	definování portů
0	definování programového čítače
1	definování střádače
2	definování stavového slova PSW
3	definování registru čítače/časovače
Q	práce s jednobitovými vstupy
H	stav hodin
0	nulování hodin
K	kmitočet vnějšího oscilátoru
I	kroková simulace
xi	bioková simulace
B	rychlá bioková simulace
W	záznam bloku progr. paměti na mgf
L	nahrání bloku progr. paměti z mgf
	cyklická změna číselné soustavy

Obsluha systému PS-48

(x znamená současný stisk symbol shift)

4.3.2. Modelování stavu mikropočítače

Klávesou „R“ se volá podprogram, který je určen pro definování obsahu jednotlivých registrů. Nejprve se zadá sada registrů, potom číslo definovaného registru a nakonec jeho nový obsah.

Klávesou „P“ se volá podprogram pro definování obsahu jednotlivých portů. Zadává se číslo portu (portu BUS odpovídá číslo 0) a jeho nový obsah.

Klávesami „0“, „1“, „2“ a „3“ se volají podprogramy pro definování programového čítače, střádače, stavového slova a registru čítače/časovače.

Návrat z těchto podprogramů se uskuteční automaticky po zadání definované hodnoty a stisku klávesy „Enter“.

Stav jednobitových testovatelných vstupů se vypíše na obrazovku po stisku klávesy „Q“. Nyní lze hodnoty na vstupech INT, T0 a T1 negovat klávesami „1“, „2“ a „3“. Návrat se provede stiskem mezerníku.

Pro definování obsahu datové paměti a externí datové paměti a k výpisu jejich obsahu se používají podprogramy, jejichž obsluha je obdobná jako u podprogramů pro definování a výpis obsahu programové paměti (kap. 4.3.1.). Místo klávesy „C“ se použije klávesa „Z“ pro práci s datovou pamětí nebo klávesa „X“ pro práci s vnější datovou pamětí.

4.3.3. Simulace mikropočítače 8048 a ladění programů

Programy, které jsou v počítači uloženy, lze pomocí simulátoru mikropočítače 8048 ladit po jednotlivých instrukcích nebo po ucelených blocích. Simulace probíhá od aktuální adresy v programovém čítači. Změnou obsahu programového čítače se změní aktuální adresa.

Krokovou simulaci vyvolá klávesa „I“. Po jejím stisku se provede simulace jedné instrukce.

Stisk kláves „I“ a „symbol shift“ současně vyvolá podprogram blokové simulace. Simulace probíhá tak dlouho, pokud není programový čítač nastaven na koncovou adresu, kterou uživatel musí zadat. Během simulace se vypisuje pouze aktuální hodnota programového čítače. Podprogram pracuje ve dvou režimech - s krokováním a bez krokování. V režimu krokování se simulace další instrukce provede po stisku libovolné klávesy. V režimu bez krokování se simulace další instrukce provede automaticky. Stisk mezerníku vyvolá návrat do hlavního programu před dosažením koncové adresy.

Odladěné úseky programu, u kterých je zaručeno dosažení koncové adresy, lze několikanásobně rychleji simulovat použitím podprogramu rychlé simulace (RYSI), který je součástí bloku interpretu. Obsluha tohoto podprogramu se vyvolává klávesou „B“.

Doba, kterou by potřeboval mikropočítač 8048 k provedení určitého úseku programu, lze zjistit podprogramem hodin. Po stisku klávesy „H“ se na obrazovku vypíše jak dlouho simulovaný mikropočítač „pracuje“. Doba je uvedena v milisekundách. Nulování hodin se provede stiskem klávesy „0“. Hodnota kmitočtu vnějšího oscilátoru se nastavuje v MHz podprogramem, který se vyvolává klávesou „K“.

4.3.4. Archivace programů

Odladěné programy lze nahrát na magnetofon a zpětně vložit do počítače.

Pro nahrání programu na magnetofon je určen podprogram vyvolávaný klávesou „W“. Zadává se počáteční a koncová adresa úseku programové paměti, který má být nahrán, a jméno programu.

Zpětné vložení programu do počítače zajišťuje podprogram, který se vyvolá klávesou „L“. Zadává se jméno požadovaného programu.

5. Ověření funkce programu na příkladech

Funkce programu byla ověřena pro všechny instrukce instrukčního souboru jednotlivě. Pro ověření funkce blo-

kové simulace byly použity podprogramy pro přenos dat z vnější paměti dat do vnitřní paměti dat, zpoždění 100 ms, násobení a dělení osmibitových čísel, sčítání čísel v BCD kódu a převod čísel z binárního kódu do kódu BCD.

Ve všech případech poskytoval program správné výsledky simulace.

Příklad 1.

Přenos bloku dat z vnější paměti dat do vnitřní paměti dat. Adresa začátku přenášeného bloku ve vnější paměti dat je v registru R0. Adresa začátku umístění bloku ve vnitřní paměti je v registru R1. V registru R2 je délka přenášeného bloku.

PC kód	návěští	instrukce
F0 80	VSTUP	MOVX A,@R0
F1 A1		MOV @R1,A
F2 18		INC R0
F3 19		INC R1
F4 EA F0		DJNZ R2,*VSTUP
F6 83		RET

Příklad 2.

Zpožďovací smyčka. Podprogram, spolu s vyvoláním instrukcí CALL 0F7h (14h,F7h), způsobí při kmitočtu vnějšího oscilátoru 3MHz zpoždění 100 ms.

PC kód	návěští	instrukce
F7 BA E5 SMS		MOV R2,#E5H
F9 BB 27		MOV R3,#27H
FB EA FB SMS1		DJNZ R2,*SMS1
FD EB FB		DJNZ R3,*SMS1
FF 83		RET

Příklad 3.

Násobení dvou osmibitových čísel. Násobenec v registru R1, násobitel v registru R2. Registr R0 je použit jako čítač smyček. Osm vyšších bitů výsledku (MSB) je uloženo ve střádači, osm nižších bitů výsledku (LSB) v registru R1.

PC kód	návěští	instrukce
100 BB08 NASOB		MOV R0,#08H
102 27		CLR A
103 97		CLR C
104 67 NAS1		RRC A
105 29		XCH A,R1
106 67		RRC A
107 29		XCH A,R1
108 E6 0B		JNC *NAS2
10A 6A		ADD A,R2

10B E8 04 NAS2	DJNZ R0,*NAS1	134 A0	MOV @R0,A
10D 67	RRC A	135 18	INC R0
10E 29	XCH A,R1	136 19	INC R1
10F 67	RRC A	137 EA 31	DJNZ R2,*PLUS1
110 29	XCH A,R1	139 83	RET
111 83	RET		

Příklad 4.

Dělení dvou osmibitových čísel. Dělenec je v registru R5. Dělitel je nepřímo adresován registrem R1 (@R1). Podíl je nepřímo adresován registrem R0 (@R0) a také je uložen ve střadači. Zbytek je uložen v registru R4.

PC kód	návěští	Instrukce
112 27	DELENI	CLR A
113 97		CLR C
114 BA 09		MOV R2,#09H
116 AC DEL1		MOV R4,A
117 FD		MOV A,R5
118 F7		RLC A
119 EA 1E		DJNZ R2,*DEL2
11B 37		CPL A
11C A0		MOV @R0,A
11D 83		RET
11E AD DEL2		MOV R5,A
11F FC		MOV A,R4
120 F7		RLC A
121 A0		MOV @R0,A
122 34 29		CALL *MINUS
124 E6 16		JNC *DEL1
126 61		ADD A,@R1
127 24 16		JMP *DEL1
129 97 MINUS		CLR C
12A F0		MOV A,@R0
12B 37		CPL A
12C 71		ADDC A,@R1
12D 37		CPL A
12E 83		RET

Příklad 5.

Sčítání čísel v BCD kódu. Adresa začátku 1. čísla je uložena v registru R0. Adresa začátku 2. čísla je uložena v registru R1. Délka čísel v bajtech je uložena v registru R2. Výsledek je uložen na místě 1. čísla.

PC kód	návěští	Instrukce
130 97	PLUS	CLR C
131 F0	PLUS1	MOV A,@R0
132 71		ADDC A,@R1
133 57		DA A

Příklad 6.

Převod šestnáctibitového čísla do kódu BCD. Nižších osm bitů (LSB) je uloženo ve střadači. Vyšších osm bitů (MSB) je uloženo v registru R2. Převoděné číslo je uloženo ve třech bajtech od adresy v registru R0.

PC kód	návěští	Instrukce
13A 28	BINBCD	XCH A,R0
13B A9		MOV R1,A
13C 28		XCH A,R0
13D BC 03		MOV R4,#03H
13F B100 BBCD1		MOV @R1,#00H
141 19		INC R1
142 EC 3F		DJNZ R4,*BBCD1
144 BB 10		MOV R3,#10H
146 97 BBCD2		CLR C
147 F7		RLC A
148 2A		XCH A,R2
149 F7		RLC A
14A 2A		XCH A,R2
14B 28		XCH A,R0
14C A9		MOV R1,A
14D 28		XCH A,R0
14E BC 03		MOV R4,#03H
150 AD		MOV R5,A
151 F1 BBCD3		MOV A,@R1
152 71		ADDC A,@R1
153 57		DA A
154 A1		MOV @R1,A
155 19		INC R1
156 EC 51		DJNZ R4,*BBCD3
158 FD		MOV A,R5
159 F6 5E		JC *BBCD4
15B EB 46		DJNZ R3,*BBCD2
15D 97		CLR C
15E 83 BBCD4		RET

6. Závěr

Programový simulátor PS-48 splňuje požadavek přenosnosti na různé typy osobních počítačů s mikroprocesory 8080 nebo Z80. Hlavní část programu, kterou tvoří podprogramy interpretu a disassembleru, je napsána ve strojovém jazyce mikroprocesoru 8080. Zbývající část, která slouží ke ko-

munikaci mezi uživatelem a programem a k obsluze strojové části, je z důvodu zachování přenosnosti na různé počítače napsána v jazyce BASIC. Program byl sestaven a očištěn na počítači ZX Spectrum a mimo jiné jej lze přenést i na počítače IQ 151 a PMD 85, které jsou značně rozšířeny ve školství. Takovýmto způsobem je možno, při využití stávajícího vybavení laboratoří, získat programový vývojový prostředek, který je vhodný nejen pro výuku programování jednočipových mikropočítačů řady MCS-48, ale i pro řešení náročnějšího aplikačního programového vybavení těchto mikropočítačů.

Program by bylo vhodné doplnit o assembler, který by ulehčil tvorbu programů, a o možnost přenášet vyvinuté programy na emulátor. Při aplikaci programu na počítači PMD 85 je možno využít programátor paměti, který se k tomuto počítači vyrábí, a doplnit obslužný podprogram.

Literatura

- [1.] Litschmann, J. : Logické obvody III. ES VSB, skripta, 1983
- [2.] Litschmann, J. : Logické obvody IV. ES VSB, skripta, 1984, str. 67-93
- [3.] Nohej, J. a kol. : Assembler 8048. Praha, Tesla ELTOS - IMA, 1984
- [4.] Nohej, J. a kol. : Základní Instrukce mikropočítače 8048. Praha, Tesla ELTOS, 1983
- [5.] Nohej, J. a kol. : Křížové programové vybavení 8048. Praha, Tesla ELTOS - IMA, 1985
- [6.] Starý, J. : Mikropočítač a jeho programování. Praha, SNTL, 1984
- [7.] Wakerly, J. F. : Mikrocomputer Architecture and programming, ruský překlad. Moskva, MIR, 1984
- [8.] Domčík, R. : Programové vybavení osobního mikropočítače umožňující tvorbu programů pro jednočipový mikropočítač 8048. Diplomová práce, VSB Ostrava 1986.
- [9.] Horák, V. : K aplikacím jednočipových mikropočítačů 8048. Amatérské rádio, 8/1985, str.302.
- [10.] Autonomní emulátor AE - 148, VTS MON Praha, 1984.
- [11.] Kolektiv autorů : Využití osobních počítačů ve výuce. ES VSB Ostrava, 1986.
- [12.] Mašková, H. : Základy programování PASCAL. Praha, SPN, 1986.
- [13.] Trpišovský, T., Zeman, V. : Emulátor TEMS 49 (příručka uživatele). Praha, Tesla ELTOS - IMA, 1985.
- [14.] Trpišovský, T., Zeman, V. : Emulátor TEMS 49 (popis rozšířené verze V - 02). Praha, Tesla ELTOS - IMA, 1986.
- [15.] Givone, D. D., Roesser, R. P. : Microprocessors/Microcomputers, ruský překlad. Moskva, MIR, 1983.

VÝPIS STROJOVÉ ČÁSTI PROGRAMU PS-48

```

E250      1      ORG #E250
          2      *D+
          3      ;*****
          4      ;** DEFINICE SYSTEMU **
          5      ;*****
          6      ZTT   DEFW 61000
          7      ZPP   DEFW #D000
          8      ZRAM  DEFW #E100
          9      ZRAMP  DEFW #E000
         10      ZTP   DEFW ZTP1
         11      TZ    DEFW #E200
         12      T0    DEFB 0
         13      T1    DEFB 0
         14      INT    DEFB 1
         15      PORT0  DEFB 0
         16      PORT1  DEFB 0
         17      PORT2  DEFB 0
         18      PORT4  DEFB 0
         19      PORT5  DEFB 0
         20      PORT6  DEFB 0
         21      PORT7  DEFB 0
         22      AKU    DEFB 0
         23      PSW    DEFB 0
         24      TIME  DEFB 0
         25      PC48   DEFW 0
         26      KONADR DEFW 0
         27      PRERUS DEFB %10000101
         28      BYTE   DEFB 0
         29      STRCYK DEFB 0
         30      HODINY  DEFW 0
         31      KONSYS DEFB 0
         32      ;*****
         33      ;** OBSLUHA **
         34      ;*****
         35      JP     DIS
         36      JP     SIMUL
         37      JP     RYSI
         38      JP     CITAC
         39      ;*****
         40      DIS    CALL VYPRTZ
         41      CALL TISKPC
         42      CALL TISKIN
         43      CALL TISKCD
         44      RET
         45      SIMUL  LD     A, (INT)
         46      OR      A
         47      JP     NZ, SIMUL1
         48      LD      A, (PRERUS)
         49      RLA
         50      JP     NC, INT03
         51      SIMUL1 CALL ADRSIM
         52      LD      A, (PRERUS)
         53      AND     %110
         54      RET     NZ
         55      LD      A, (STRCYK)
         56      AND     #E0
         57      RET     Z
         58      LD      A, (STRCYK)
         59      AND     #1F
         60      LD      (STRCYK), A
         61      CITAC1 LD     HL, TIME
         62      INC     (HL)
         63      RET     NZ
         64      LD      A, (PRERUS)
         65      OR      %100000
         66      LD      (PRERUS), A
         67      RRA
         68      RET     C
         69      CALL ZASPC
         70      CALL STAV2
         71      LD      HL, 7
         72      LD      (PC48), HL
         73      RET
         74      INT03  CALL ZASPC
         75      CALL STAV2
         76      LD      HL, 3
         77      LD      (PC48), HL
         78      RET
         79      RYSI   CALL DIS
         80      CALL SIMUL
         81      CALL KONTR
         82      JP     RYSI
         83      KONTR LD     HL, KONADR

```

```

580893A69E2 84      LD     A, (PC48)
58092BE      85      CP     (HL)
58093C0      86      RET     NZ
5809423      87      INC     HL
580953A6AE2 88      LD     A, (PC48+1)
58098BE      89      CP     (HL)
58099C0      90      RET     NZ
58100E1      91      POP     HL
58101C9      92      RET
581023A6DE2 93      CITAC LD     A, (PRERUS)
58105E606    94      AND     %110
58107FE02    95      CP     %010
58109C0      96      RET     NZ
58110C3B1E2 97      JP     CITAC1
          98      ;*****
          99      ;** DISASSEMBLER **
         100      ;*****
         101      ASCI    LD     D, A
         102      AND     #0F
         103      CALL PREVOD
         104      LD      E, A
         105      LD      A, D
         106      AND     #F0
         107      RLCA
         108      RLCA
         109      RLCA
         110      RLCA
         111      CALL PREVOD
         112      LD      D, A
         113      RET
         114      PREVOD ADD    A, #30
         115      CP     #3A
         116      RET     M
         117      ADD     A, 7
         118      RET
         119      ADRKOD LD     HL, (PC48)
         120      EX      DE, HL
         121      LD      HL, (ZPP)
         122      ADD     HL, DE
         123      LD      A, (HL)
         124      RET
         125      ADRTEX CALL ADRKOD
         126      LD      HL, (ZTT)
         127      LD      C, A
         128      LD      B, 0
         129      LD      D, 9
         130      L1     ADD     HL, BC
         131      DEC     D
         132      JP     NZ, L1
         133      RET
         134      TISKPC LD     HL, (PC48)
         135      LD      A, H
         136      CALL ASCI
         137      LD      B, D
         138      LD      C, E
         139      LD      A, L
         140      CALL ASCI
         141      LD      HL, (TZ)
         142      LD      (HL), B
         143      INC     HL
         144      LD      (HL), C
         145      INC     HL
         146      LD      (HL), D
         147      INC     HL
         148      LD      (HL), E
         149      RET
         150      TISKIN CALL ADRTEX
         151      EX      DE, HL
         152      LD      HL, (TZ)
         153      LD      A, L
         154      ADD     A, #A
         155      LD      L, A
         156      LD      B, #09
         157      L3     LD      A, (DE)
         158      CP     #80
         159      CALL P, ZONA2
         160      CP     #"
         161      JP     Z, DATA1
         162      LD      (HL), A
         163      INC     HL
         164      INC     DE
         165      DEC     B
         166      JP     NZ, L3
         167      SCF
         168      CCF
         169      RET
         170      ZONA2  AND     %01111111
         171      INC     HL
         172      RET
         173      DATA1 PUSH   HL

```

58230CD1CE3	174	CALL	ADRKOD	58375C304E4	264	JP	L8
5823323	175	INC	HL	583783A6EE2	265	STAV1	LD A, (BYTE)
582347E	176	LD	A, (HL)	583812A69E2	266	LD	HL, (PC48)
58235CD01E3	177	CALL	ASCII	5838423	267	L7	INC HL
58238E1	178	POP	HL	583853D	268	DEC	A
5823972	179	LD	(HL), D	58386C210E4	269	JP	NZ, L7
5824023	180	INC	HL	583892269E2	270	LD	(PC48), HL
5824173	181	LD	(HL), E	58392216FE2	271	STAV3	LD HL, STRCYK
5824223	182	INC	HL	5839534	272	INC	(HL)
582433648	183	LD	(HL), "H"	583962170E2	273	LD	HL, HODINY
5824537	184	SCF		5839934	274	INC	(HL)
58246C9	185	RET		58400C0	275	RET	NZ
582470601	186	TISKCD	LD B, 01	5840123	276	INC	HL
58249D28DE3	187	JP	NC, L4	5840234	277	INC	(HL)
5825204	188	INC	B	58403C0	278	RET	NZ
5825378	189	L4	LD A, B	5840423	279	INC	HL
58254326EE2	190	LD	(BYTE), A	5840534	280	INC	(HL)
58257CD1CE3	191	CALL	ADRKOD	58406C9	281	RET	
58260CD01E3	192	CALL	ASCII	58407CD18E4	282	STAV2	CALL STAV3
582632A5AE2	193	LD	HL, (TZ)	58410C30AE4	283	JP	STAV1
582667D	194	LD	A, L	58413CDD7E3	284	MAR7	CALL REGIST
58267C605	195	ADD	A, 05	584167E	285	MAR71	LD A, (HL)
582696F	196	L5	LD L, A	584173266E2	286	LD	(AKU), A
5827072	197	LD	(HL), D	58420C30AE4	287	JP	STAV1
5827123	198	INC	HL	58423CDD7E3	288	MR7A	CALL REGIST
5827273	199	LD	(HL), E	584263A66E2	289	MR7A1	LD A, (AKU)
5827305	200	DEC	B	5842977	290	LD	(HL), A
58274C8	201	RET	Z	58430C30AE4	291	JP	STAV1
58275CD1CE3	202	CALL	ADRKOD	58433CDEBE3	292	MAa	CALL REGa
5827823	203	INC	HL	58436C330E4	293	JP	MAR71
582797E	204	LD	A, (HL)	58439CDEBE3	294	MaA	CALL REGa
58280CD01E3	205	CALL	ASCII	58442C33AE4	295	JP	MR7A1
582832A5AE2	206	LD	HL, (TZ)	58445CDF6E3	296	MR7t	CALL DATA
582867D	207	LD	A, L	58448CDD7E3	297	CALL	REGIST
58287C607	208	ADD	A, #07	5845170	298	MR7t1	LD (HL), B
58289C39DE3	209	JP	L5	58452C327E4	299	JP	STAV2
582923E20	210	VYPRTZ	LD A, #20	58455CDF6E3	300	Mat	CALL DATA
582942A5AE2	211	LD	HL, (TZ)	58458CDEBE3	301	CALL	REGa
58297061E	212	LD	B, 30	58461C353E4	302	JP	MR7t1
5829977	213	L2	LD (HL), A	58464CDF6E3	303	Mat	CALL DATA
5830023	214	INC	HL	584673266E2	304	Mat1	LD (AKU), A
5830105	215	DEC	B	58470C327E4	305	JP	STAV2
58302C28BE3	216	JP	NZ, L2	584733A67E2	306	MAPSW	LD A, (PSW)
58305C9	217	RET		584763266E2	307	LD	(AKU), A
	218	;*****					
	219	;*** INTERPRET ***					
	220	;*****					
58306CD1CE3	221	ADRSIM	CALL ADRKOD	58479C30AE4	308	JP	STAV1
583091600	222	LD	D, 0	584823A66E2	309	MPSWA	LD A, (AKU)
583114F	223	LD	C, A	58485F608	310	OR	%1000
5831287	224	ADD	A, A	584873267E2	311	LD	(PSW), A
58313D2CDE3	225	JP	NC, L10	58490C30AE4	312	JP	STAV1
5831614	226	INC	D	584933A68E2	313	MAT	LD A, (TIME)
583175F	227	L10	LD E, A	584963266E2	314	LD	(AKU), A
583182A58E2	228	LD	HL, (ZTP)	58499C30AE4	315	JP	STAV1
5832119	229	ADD	HL, DE	585023A66E2	316	MTA	LD A, (AKU)
583225E	230	LD	E, (HL)	585053268E2	317	LD	(TIME), A
5832323	231	INC	HL	58508C30AE4	318	JP	STAV1
5832456	232	LD	D, (HL)	58511CDD7E3	319	MXAa	CALL REGIST
58325EB	233	EX	DE, HL	585147E	320	LD	A, (HL)
58326E9	234	JP	(HL)	585152A56E2	321	LD	HL, (ZRAM)
5832779	235	REGIST	LD A, C	585186F	322	LD	L, A
58328E607	236	AND	%111	585197E	323	LD	A, (HL)
583304F	237	LD	C, A	58520325FE2	324	LD	(PORT0), A
583313A67E2	238	LD	A, (PSW)	58523C363E4	325	JP	Mat1
58334E610	239	AND	%10000	58526CDD7E3	326	MXAa	CALL REGIST
58336CAE5E3	240	JP	Z, L6	585297E	327	LD	A, (HL)
583393E18	241	LD	A, 24	585302A56E2	328	LD	HL, (ZRAM)
5834181	242	L6	ADD A, C	585336F	329	LD	L, A
583422A54E2	243	LD	HL, (ZRAM)	585343A66E2	330	LD	A, (AKU)
583456F	244	LD	L, A	58537325FE2	331	LD	(PORT0), A
58346C9	245	RET		58540C366E5	332	JP	ANLPt1
58347CDD7E3	246	REGa	CALL REGIST	58543CD1CE3	333	MPAa	CALL ADRKOD
583507E	247	LD	A, (HL)	585463A66E2	334	LD	A, (AKU)
58351E63F	248	AND	%111111	585496F	335	LD	L, A
583532A54E2	249	LD	HL, (ZRAM)	585507E	336	LD	A, (HL)
583566F	250	LD	L, A	58551C363E4	337	JP	Mat1
58357C9	251	RET		585542A52E2	338	MP3AaA	LD HL, (ZPP)
58358CD1CE3	252	DATA	CALL ADRKOD	585571603	339	LD	D, 3
583614F	253	LD	C, A	585593A66E2	340	LD	A, (AKU)
5836223	254	INC	HL	585625F	341	LD	E, A
583637E	255	LD	A, (HL)	5856319	342	ADD	HL, DE
5836446	256	LD	B, (HL)	585647E	343	LD	A, (HL)
58365C9	257	RET		58565C363E4	344	JP	Mat1
5836679	258	P47	LD A, C	58568CDFEE3	345	MDP47A	CALL P47
58367E603	259	AND	%11	585713A66E2	346	LD	A, (AKU)
583692162E2	260	LD	HL, PORT4	58574E60F	347	AND	%1111
583723D	261	L8	DEC A	5857677	348	LD	(HL), A
58373F8	262	RET	M	5857747	349	L9	LD B, A
5837423	263	INC	HL	585783A61E2	350	LD	A, (PORT2)
				58581B0	351	OR	B
				585823261E2	352	LD	(PORT2), A
				58585C327E4	353	JP	STAV2

58588CDFEE3	354	MDAP47	CALL	P47	58790C343E4	444		JP	Mat1
585917E	355		LD	A, (HL)	58793CDFEE3	445	ADP47A	CALL	P47
58592E60F	356		AND	%1111	587963A66E2	446		LD	A, (AKU)
585943266E2	357		LD	(AKU), A	58799A6	447		AND	(HL)
58597C3D1E4	358		JP	L9	58800C3D0E4	448		JP	L9-1
58600CDD7E3	359	XCHAR7	CALL	REGIST	58803CDFEE3	449	ODP47A	CALL	P47
586033A66E2	360	XCHR1	LD	A, (AKU)	588063A66E2	450		LD	A, (AKU)
5860647	361		LD	B, A	58809E60F	451		AND	#0F
586077E	362		LD	A, (HL)	58811B6	452		OR	(HL)
5860870	363		LD	(HL), B	58812C3D0E4	453		JP	L9-1
586093266E2	364	XCHR2	LD	(AKU), A	588153A66E2	454	RRA	LD	A, (AKU)
58612C30AE4	365		JP	STAV1	588180F	455		RRCA	
58615CDEBE3	366	XCHAA	CALL	REGa	58819C3F1E4	456		JP	XCHR2
58618C3EBE4	367		JP	XCHR1	588223A66E2	457	RLA	LD	A, (AKU)
58621CDEBE3	368	XDAa	CALL	REGa	5882507	458		RLCA	
586247E	369		LD	A, (HL)	58826C3F1E4	459		JP	XCHR2
5862547	370		LD	B, A	588293A67E2	460	RRCA	LD	A, (PSW)
58626E6F0	371		AND	#F0	5883217	461		RLA	
5862877	372		LD	(HL), A	5883347	462		LD	B, A
586293A66E2	373		LD	A, (AKU)	588343A66E2	463		LD	A, (AKU)
586324F	374		LD	C, A	588371F	464		RRA	
58633E60F	375		AND	#0F	588383266E2	465	RRCA1	LD	(AKU), A
58635B6	376		OR	(HL)	588417B	466		LD	A, B
5863677	377		LD	(HL), A	588421F	467		RRA	
586377B	378		LD	A, B	588433267E2	468		LD	(PSW), A
58638E60F	379		AND	#0F	58846C30AE4	469		JP	STAV1
5864047	380		LD	B, A	588493A67E2	470	RLCA	LD	A, (PSW)
5864179	381		LD	A, C	5885217	471		RLA	
58642E6F0	382		AND	#F0	5885347	472		LD	B, A
58644B0	383		OR	B	588543A66E2	473		LD	A, (AKU)
58645C3F1E4	384		JP	XCHR2	5885717	474		RLA	
586483A66E2	385	SWAP	LD	A, (AKU)	58858C3D6E5	475		JP	RRCA1
586510F	386		RRCA		588612166E2	476	INCA	LD	HL, AKU
586520F	387		RRCA		5886434	477	INCA1	INC	(HL)
586530F	388		RRCA		58865C30AE4	478		JP	STAV1
586540F	389		RRCA		58868CDD7E3	479	INCR7	CALL	REGIST
58655C3F1E4	390		JP	XCHR2	58871C3F0E5	480		JP	INCA1
5865879	391	P02	LD	A, C	58874CDEBE3	481	INCA	CALL	REGa
58659E603	392		AND	%011	58877C3F0E5	482		JP	INCA1
58661215FE2	393		LD	HL, PORT0	588802166E2	483	DECA	LD	HL, AKU
58664C304E4	394		JP	L8	5888335	484	DECA1	DEC	(HL)
58667CD22E5	395	INPUT	CALL	P02	58884C30AE4	485		JP	STAV1
586707E	396		LD	A, (HL)	58887CDD7E3	486	DECR7	CALL	REGIST
58671C363E4	397		JP	Mat1	58890C303E6	487		JP	DECA1
586743A66E2	398	OUTBA	LD	A, (AKU)	58893CDF6E3	488	DJNZRt	CALL	DATA
58677325FE2	399		LD	(PORT0), A	58896CD27E4	489		CALL	STAV2
58680C327E4	400		JP	STAV2	58899CDD7E3	490		CALL	REGIST
58683CD22E5	401	OUTPA	CALL	P02	5890235	491		DEC	(HL)
586863A66E2	402		LD	A, (AKU)	58903C8	492		RET	Z
58689C366E5	403		JP	ANLPt1	589042A69E2	493		LD	HL, (PC4B)
58692CDD7E3	404	ANAR7	CALL	REGIST	589072B	494		DEC	HL
586953A66E2	405	ANAR1	LD	A, (AKU)	589086B	495		LD	L, B
58698A6	406		AND	(HL)	589092269E2	496		LD	(PC4B), HL
58699C3F1E4	407		JP	XCHR2	58912C9	497		RET	
58702CDEBE3	408	ANAA	CALL	REGa	58913F5	498	PSWCAC	PUSH	AF
58705C347E5	409		JP	ANAR1	58914C1	499		POP	BC
58708CDF6E3	410	Ant	CALL	DATA	5891579	500		LD	A, C
587113A66E2	411		LD	A, (AKU)	589161F	501		RRA	
58714A6	412		AND	(HL)	589173A67E2	502		LD	A, (PSW)
58715C363E4	413		JP	Mat1	5892017	503		RLA	
58718CDF6E3	414	ANLPt	CALL	DATA	5892147	504		LD	B, A
58721CD22E5	415		CALL	P02	5892279	505		LD	A, C
587247E	416		LD	A, (HL)	5892317	506		RLA	
58725A0	417		AND	B	5892417	507		RLA	
5872677	418	ANLPt1	LD	(HL), A	5892517	508		RLA	
58727C327E4	419		JP	STAV2	5892617	509		RLA	
58730CDD7E3	420	ORLAR7	CALL	REGIST	589277B	510		LD	A, B
587333A66E2	421	ORLAR1	LD	A, (AKU)	5892817	511		RLA	
58736B6	422		OR	(HL)	589290F	512		RRCA	
58737C3F1E4	423		JP	XCHR2	589300F	513		RRCA	
58740CDEBE3	424	ORLAa	CALL	REGa	589313267E2	514		LD	(PSW), A
58743C36DE5	425		JP	ORLAR1	58934C9	515		RET	
58746CDF6E3	426	ORLat	CALL	DATA	589353A67E2	516	CACPSW	LD	A, (PSW)
587493A66E2	427		LD	A, (AKU)	5893847	517		LD	B, A
58752B6	428		OR	(HL)	589391F	518		RRA	
58753C363E4	429		JP	Mat1	589401F	519		RRA	
58756CDF6E3	430	ORPt	CALL	DATA	589411F	520		RRA	
58759CD22E5	431		CALL	P02	589424F	521		LD	C, A
587627E	432		LD	A, (HL)	589437B	522		LD	A, B
58763B0	433		OR	B	5894417	523		RLA	
58764C366E5	434		JP	ANLPt1	5894579	524		LD	A, C
58767CDD7E3	435	XRAR7	CALL	REGIST	5894617	525		RLA	
587703A66E2	436	XRAR1	LD	A, (AKU)	589474F	526		LD	C, A
58773AE	437		XOR	(HL)	58948C5	527		PUSH	BC
58774C3F1E4	438		JP	XCHR2	58949F1	528		POP	AF
58777CDEBE3	439	XRAa	CALL	REGa	58950C9	529		RET	
58780C392E5	440		JP	XRAR1	589513A67E2	530	CARRY0	LD	A, (PSW)
58783CDF6E3	441	XRAt	CALL	DATA	58954E67F	531		AND	%01111111
587863A66E2	442		LD	A, (AKU)	589563267E2	532		LD	(PSW), A
58789AE	443		XOR	(HL)	58959C30AE4	533		JP	STAV1

58962CDD7E3	534	ADAR7	CALL	REGIST	59159CD0AE4	624	STATT	CALL	STAV1
589653A66E2	535	ADAR1	LD	A, (AKU)	59162216FE2	625	LD	HL,STRCYK	
5896886	536		ADD	A, (HL)	591653600	626	LD	(HL),0	
589693266E2	537		LD	(AKU),A	591673A6DE2	627	LD	A, (PRERUS)	
58972CD21E6	538		CALL	PSWCAC	59170E6F9	628	AND	%11111001	
58975C30AE4	539		JP	STAV1	59172326DE2	629	LD	(PRERUS),A	
58978CDEBE3	540	ADAA	CALL	REGa	59175C9	630	RET		
58981C355E6	541		JP	ADAR1	591763A6DE2	631	STATC	LD	A, (PRERUS)
58984CDF6E3	542	ADAt	CALL	DATA	59179E6FB	632	AND	%11111011	
589873A66E2	543		LD	A, (AKU)	59181F602	633	OR	%10	
5899086	544		ADD	A, (HL)	59183C304E7	634	JP	INTER1	
589913266E2	545		LD	(AKU),A	591863A6DE2	635	STOPTC	LD	A, (PRERUS)
58994CD21E6	546		CALL	PSWCAC	59189F604	636	OR	%100	
58997C327E4	547		JP	STAV2	59191C304E7	637	JP	INTER1	
59000CDD7E3	548	ADCAR7	CALL	REGIST	591943A6DE2	638	T0HOD	LD	A, (PRERUS)
59003CD37E6	549	ADCAR1	CALL	CACPSW	59197F610	639	OR	%10000	
590063A66E2	550		LD	A, (AKU)	59199C304E7	640	JP	INTER1	
590098E	551		ADC	A, (HL)	592023A6DE2	641	SELMB0	LD	A, (PRERUS)
590103266E2	552		LD	(AKU),A	59205E6F7	642	AND	%11111011	
59013CD21E6	553		CALL	PSWCAC	59207C304E7	643	JP	INTER1	
59016C30AE4	554	NIC	JP	STAV1	592103A6DE2	644	SELMB1	LD	A, (PRERUS)
59019CDEBE3	555	ADCAa	CALL	REGa	59213F608	645	OR	%1000	
59022C37BE6	556		JP	ADCAR1	59215C304E7	646	JP	INTER1	
59025CDF6E3	557	ADCAt	CALL	DATA	592183A67E2	647	SELRB0	LD	A, (PSW)
59028CD37E6	558		CALL	CACPSW	59221E6EF	648	AND	%11101111	
590313A66E2	559		LD	A, (AKU)	59223C3AEE6	649	JP	NAS1	
590348E	560		ADC	A, (HL)	592263A67E2	650	SELRB1	LD	A, (PSW)
590353266E2	561		LD	(AKU),A	59229F610	651	OR	%10000	
59038CD21E6	562		CALL	PSWCAC	59231C3AEE6	652	JP	NAS1	
59041C327E4	563		JP	STAV2	592342A54E2	653	ZASOB	LD	HL, (ZRAM)
590443E00	564	CLRA	LD	A,0	592373A67E2	654	LD	A, (PSW)	
59046C3F1E4	565		JP	XCHR2	59240E607	655	AND	%111	
590493A67E2	566	NAS0F0	LD	A, (PSW)	5924207	656	RLCA		
59052E6DF	567		AND	%11011111	592430608	657	LD	B,8	
590543267E2	568	NAS1	LD	(PSW),A	5924580	658	ADD	A,B	
59057C30AE4	569		JP	STAV1	592466F	659	LD	L,A	
590603A6DE2	570	NAS1F0	LD	A, (PRERUS)	59247C9	660	RET		
59063E6BF	571		AND	%10111111	59248CDF6E3	661	PODPA	CALL	DATA
59065326DE2	572		LD	(PRERUS),A	5925158	662	LD	E,B	
59068C30AE4	573		JP	STAV1	5925279	663	LD	A,C	
590713A66E2	574	CPLA	LD	A, (AKU)	5925307	664	RLCA		
590742F	575		CPL		5925407	665	RLCA		
59075C3F1E4	576		JP	XCHR2	5925507	666	RLCA		
590783A67E2	577	CARRYN	LD	A, (PSW)	59256E607	667	AND	%111	
5908117	578		RLA		5925857	668	LD	D,A	
590823F	579		CCF		592593A6DE2	669	LD	A, (PRERUS)	
590831F	580		RRA		59262E608	670	AND	%1000	
59084C3AEE6	581		JP	NAS1	5926482	671	ADD	A,D	
590873A67E2	582	NAS0FN	LD	A, (PSW)	5926557	672	LD	D,A	
5909017	583		RLA		59266C9	673	RET		
5909117	584		RLA		592672A69E2	674	ZASPC	LD	HL, (PC4B)
5909217	585		RLA		592703A67E2	675	LD	A, (PSW)	
590933F	586		CCF		59273E6F0	676	AND	#F0	
590941F	587		RRA		5927547	677	LD	B,A	
590951F	588		RRA		592767C	678	LD	A,H	
590961F	589		RRA		59277E60F	679	AND	#0F	
59097C3AEE6	590		JP	NAS1	5927980	680	ADD	A,B	
591003A6DE2	591	NAS1FN	LD	A, (PRERUS)	5928067	681	LD	H,A	
5910317	592		RLA		59281EB	682	EX	DE,HL	
5910417	593		RLA		59282CD62E7	683	CALL	ZASOB	
591053F	594		CCF		5928573	684	LD	(HL),E	
591061F	595		RRA		5928623	685	INC	HL	
591071F	596		RRA		5928772	686	LD	(HL),D	
59108326DE2	597		LD	(PRERUS),A	592882167E2	687	LD	HL,PSW	
59111C30AE4	598		JP	STAV1	5929134	688	INC	(HL)	
59114CD37E6	599	DAA	CALL	CACPSW	59292C9	689	RET		
591173A66E2	600		LD	A, (AKU)	59293CD27E4	690	PODPA	CALL	STAV2
5912027	601		DAA		59296CD83E7	691	CALL	ZASPC	
591213266E2	602		LD	(AKU),A	59299CDDEE7	692	CALL	PC4B2M	
59124CD21E6	603		CALL	PSWCAC	59302CD70E7	693	CALL	PODPA	
59127C30AE4	604		JP	STAV1	59305EB	694	EX	DE,HL	
5913079	605	INTER	LD	A,C	593062269E2	695	LD	(PC4B),HL	
5913117	606		RLA		59309C9	696	RET		
5913217	607		RLA		59310CD27E4	697	NAVR	CALL	STAV2
5913317	608		RLA		593132167E2	698	LD	HL,PSW	
5913417	609		RLA		5931635	699	DEC	(HL)	
591353A6DE2	610		LD	A, (PRERUS)	59317CD62E7	700	CALL	ZASOB	
5913817	611		RLA		593205E	701	LD	E, (HL)	
591390F	612		RRCA		5932123	702	INC	HL	
59140326DE2	613	INTER1	LD	(PRERUS),A	593227E	703	LD	A, (HL)	
59143C30AE4	614		JP	STAV1	59323E60F	704	AND	#0F	
5914679	615	TCNTI	LD	A,C	5932557	705	LD	D,A	
5914717	616		RLA		59326EB	706	EX	DE,HL	
5914817	617		RLA		593272269E2	707	LD	(PC4B),HL	
5914917	618		RLA		5933079	708	LD	A,C	
5915017	619		RLA		59331FE83	709	CP	#B3	
591513A6DE2	620		LD	A, (PRERUS)	59333C8	710	RET	Z	
591541F	621		RRA		593341A	711	LD	A, (DE)	
5915507	622		RLCA		59335E6F0	712	AND	#F0	
59156C304E7	623		JP	INTER1	5933747	713	LD	B,A	

593383A67E2	714	LD	A, (PSW)	59541FA9CE8	804	JP	M, L30
59341E60F	715	AND	#0F	5954407	805	RLCA	
5934380	716	ADD	A, B	59545C394E8	806	JP	L25
593443267E2	717	LD	(PSW), A	595485F	807	LD	E, A
59347C9	718	RET		595493A66E2	808	LD	A, (AKU)
59348CD70E7	719	SKOK	CALL PODFA	59552A3	809	AND	E
59351EB	720	EX	DE, HL	59553C8	810	RET	Z
593522269E2	721	LD	(PC48), HL	59554C3F2E7	811	JP	PLSKOK
59355CD27E4	722	CALL	STAV2		812	;*****	
593582A69E2	723	PC482M	LD HL, (PC48)		813	;** TABULKA **	
593612B	724	DEC	HL		814	;** SKOKU SIMULACE **	
593622B	725	DEC	HL		815	;*****	
593632269E2	726	LD	(PC48), HL	595570AE4	816	ZTP1	DEFW STAV1
59366C9	727	RET		5955988E6	817	DEFW	NIC
59367CD1CE3	728	SKOKa	CALL ADRKOD	5956132E5	818	DEFW	OUTBA
593703A66E2	729	LD	A, (AKU)	5956368E6	819	DEFW	ADAT
593736F	730	LD	L, A	59565D4E7	820	DEFW	SKOK
5937446	731	LD	B, (HL)	59567FAE6	821	DEFW	INTER
59375CD27E4	732	CALL	STAV2	5956988E6	822	DEFW	NIC
593782A69E2	733	PLSKOK	LD HL, (PC48)	5957100E6	823	DEFW	DECA
593812B	734	DEC	HL	595732BE5	824	DEFW	INPUT
5938268	735	LD	L, B	595752BE5	825	DEFW	INPUT
593832269E2	736	LD	(PC48), HL	595772BE5	826	DEFW	INPUT
59386C9	737	RET		5957988E6	827	DEFW	NIC
59387CDF6E3	738	SKOK5	CALL DATA	59581DCE4	828	DEFW	MDAP47
59390CD27E4	739	CALL	STAV2	59583DCE4	829	DEFW	MDAP47
59393C9	740	RET		59585DCE4	830	DEFW	MDAP47
59394CDFBE7	741	SKOKC1	CALL SKOK5	59587DCE4	831	DEFW	MDAP47
593973A67E2	742	LD	A, (PSW)	59589FAE5	832	DEFW	INCa
5940017	743	RLA		59591FAE5	833	DEFW	INCa
59401D0	744	RET	NC	5959388E8	834	DEFW	SKOKB
59402C3F2E7	745	JP	PLSKOK	5959591E6	835	DEFW	ADCa
59405CDFBE7	746	SKOKC0	CALL SKOK5	595979DE7	836	DEFW	PODP
594083A67E2	747	LD	A, (PSW)	59599FAE6	837	DEFW	INTER
5941117	748	RLA		596016AE8	838	DEFW	SKOKTF
59412D8	749	RET	C	59603EDE5	839	DEFW	INCA
59413C3F2E7	750	JP	PLSKOK	59605F4E5	840	DEFW	INCR7
59416CDFBE7	751	SKOKZ0	CALL SKOK5	59607F4E5	841	DEFW	INCR7
594193A66E2	752	LD	A, (AKU)	59609F4E5	842	DEFW	INCR7
59422B7	753	SKOKZ5	OR A	59611F4E5	843	DEFW	INCR7
59423C0	754	RET	NZ	59613F4E5	844	DEFW	INCR7
59424C3F2E7	755	JP	PLSKOK	59615F4E5	845	DEFW	INCR7
59427CDFBE7	756	SKOKZ1	CALL SKOK5	59617F4E5	846	DEFW	INCR7
594303A66E2	757	LD	A, (AKU)	59619F4E5	847	DEFW	INCR7
59433B7	758	SKOKZ6	OR A	59621F7E4	848	DEFW	XCHAA
59434C8	759	RET	Z	59623F7E4	849	DEFW	XCHAA
59435C3F2E7	760	JP	PLSKOK	5962588E6	850	DEFW	NIC
59438CDFBE7	761	SKOKF0	CALL SKOK5	5962760E4	851	DEFW	MAT
594413A67E2	762	LD	A, (PSW)	59629D4E7	852	DEFW	SKOK
59444E620	763	AND	%100000	596310AE7	853	DEFW	TCNTI
59446C8	764	RET	Z	596334FE8	854	DEFW	SK00T0
59447C3F2E7	765	JP	PLSKOK	59635A4E6	855	DEFW	CLRA
59450CDFBE7	766	SKOKF1	CALL SKOK5	59637E8E4	856	DEFW	XCHAR7
594533A6DE2	767	LD	A, (PRERUS)	59639E8E4	857	DEFW	XCHAR7
59456E640	768	AND	%1000000	59641E8E4	858	DEFW	XCHAR7
59458C8	769	RET	Z	59643E8E4	859	DEFW	XCHAR7
59459C3F2E7	770	JP	PLSKOK	59645E8E4	860	DEFW	XCHAR7
59462CDFBE7	771	SK00T1	CALL SKOK5	59647E8E4	861	DEFW	XCHAR7
594653A5CE2	772	LD	A, (T0)	59649E8E4	862	DEFW	XCHAR7
59468C329E8	773	JP	SKOKZ6	59651E8E4	863	DEFW	XCHAR7
59471CDFBE7	774	SK00T0	CALL SKOK5	59653FDE4	864	DEFW	XDAa
594743A5CE2	775	LD	A, (T0)	59655FDE4	865	DEFW	XDAa
59477C31EE8	776	JP	SKOKZ5	5965788E8	866	DEFW	SKOKB
59480CDFBE7	777	SK01T1	CALL SKOK5	5965988E6	867	DEFW	NIC
594833A5DE2	778	LD	A, (T1)	596619DE7	868	DEFW	PODP
59486C329E8	779	JP	SKOKZ6	596630AE7	869	DEFW	TCNTI
59489CDFBE7	780	SK01T0	CALL SKOK5	5966546E8	870	DEFW	SK00T1
594923A5DE2	781	LD	A, (T1)	59667BFE6	871	DEFW	CPLA
59495C31EE8	782	JP	SKOKZ5	5966988E6	872	DEFW	NIC
59498CDFBE7	783	SKOKTF	CALL SKOK5	596713BE5	873	DEFW	OUTPA
595013A6DE2	784	LD	A, (PRERUS)	596733BE5	874	DEFW	OUTPA
59504E620	785	AND	#20	5967588E6	875	DEFW	NIC
59506B7	786	OR	A	59677C8E4	876	DEFW	MDP47A
59507C8	787	RET	Z	59679C8E4	877	DEFW	MDP47A
595083A6DE2	788	LD	A, (PRERUS)	59681C8E4	878	DEFW	MDP47A
59511E6DF	789	AND	%11011111	59683C8E4	879	DEFW	MDP47A
59513326DE2	790	LD	(PRERUS), A	5968574E5	880	DEFW	ORLAa
59516C3F2E7	791	JP	PLSKOK	5968774E5	881	DEFW	ORLAa
59519CDFBE7	792	SKOKI	CALL SKOK5	596897DE4	882	DEFW	MAT
595223A5EE2	793	LD	A, (INT)	596917AE5	883	DEFW	ORLAa
59525C31EE8	794	JP	SKOKZ5	59693D4E7	884	DEFW	SKOK
59528CDFBE7	795	SKOKB	CALL SKOK5	5969528E7	885	DEFW	STATC
5953179	796	LD	A, C	5969761E8	886	DEFW	SK01T0
5953207	797	RLCA		5969918E5	887	DEFW	SWAP
5953307	798	RLCA		597016AE5	888	DEFW	ORLAR7
5953407	799	RLCA		597036AE5	889	DEFW	ORLAR7
59535E607	800	AND	%111	597056AE5	890	DEFW	ORLAR7
595375F	801	LD	E, A	597076AE5	891	DEFW	ORLAR7
595383E01	802	LD	A, 1	597096AE5	892	DEFW	ORLAR7
595401D	803	L25	DEC E	597116AE5	893	DEFW	ORLAR7

597136AE5	894	DEFW ORLAR7	5989337E4	984	DEFW MR7A
597156AE5	895	DEFW ORLAR7	5989537E4	985	DEFW MR7A
597174EE5	896	DEFW ANAa	5989737E4	986	DEFW MR7A
597194EE5	897	DEFW ANAa	5989937E4	987	DEFW MR7A
5972188E8	898	DEFW SKOKB	5990137E4	988	DEFW MR7A
5972354E5	899	DEFW ANt	5990337E4	989	DEFW MR7A
597259DE7	900	DEFW PDDP	5990537E4	990	DEFW MR7A
5972717E7	901	DEFW STATT	5990737E4	991	DEFW MR7A
5972958E8	902	DEFW SK01T1	5990957E4	992	DEFW Mat
59731EAE6	903	DEFW DAA	5991157E4	993	DEFW Mat
5973344E5	904	DEFW ANAR7	5991388E8	994	DEFW SKOKB
5973544E5	905	DEFW ANAR7	59915E7E7	995	DEFW SKOKa
5973744E5	906	DEFW ANAR7	599179DE7	996	DEFW PDDP
5973944E5	907	DEFW ANAR7	59919DCE6	997	DEFW NAS1FN
5974144E5	908	DEFW ANAR7	599212EE8	998	DEFW SKOKF0
5974344E5	909	DEFW ANAR7	5992388E6	999	DEFW NIC
5974544E5	910	DEFW ANAR7	599254DE4	1000	DEFW MR7t
5974744E5	911	DEFW ANAR7	599274DE4	1001	DEFW MR7t
5974962E6	912	DEFW ADAa	599294DE4	1002	DEFW MR7t
5975162E6	913	DEFW ADAa	599314DE4	1003	DEFW MR7t
5975386E4	914	DEFW MTA	599334DE4	1004	DEFW MR7t
5975588E6	915	DEFW NIC	599354DE4	1005	DEFW MR7t
5975704E7	916	DEFW SKOK	599374DE4	1006	DEFW MR7t
5975932E7	917	DEFW STOPTC	599394DE4	1007	DEFW MR7t
5976188E6	918	DEFW NIC	5994188E6	1008	DEFW NIC
59763CDE5	919	DEFW RRCA	5994388E6	1009	DEFW NIC
5976552E6	920	DEFW ADAR7	5994588E6	1010	DEFW NIC
5976752E6	921	DEFW ADAR7	5994788E6	1011	DEFW NIC
5976952E6	922	DEFW ADAR7	59949D4E7	1012	DEFW SKOK
5977152E6	923	DEFW ADAR7	5995152E7	1013	DEFW SELRB0
5977352E6	924	DEFW ADAR7	5995318E8	1014	DEFW SKOKZ0
5977552E6	925	DEFW ADAR7	5995569E4	1015	DEFW MAPSW
5977752E6	926	DEFW ADAR7	5995707E6	1016	DEFW DECR7
5977952E6	927	DEFW ADAR7	5995907E6	1017	DEFW DECR7
5978188E6	928	DEFW ADCaA	5996107E6	1018	DEFW DECR7
5978388E6	929	DEFW ADCaA	5996307E6	1019	DEFW DECR7
5978588E8	930	DEFW SKOKB	5996507E6	1020	DEFW DECR7
5978788E6	931	DEFW NIC	5996707E6	1021	DEFW DECR7
597899DE7	932	DEFW PDDP	5996907E6	1022	DEFW DECR7
597913AE7	933	DEFW T0H0D	5997107E6	1023	DEFW DECR7
597933AE8	934	DEFW SKOKF1	5997399E5	1024	DEFW XRAa
59795BF5	935	DEFW RRA	5997599E5	1025	DEFW XRAa
5979778E6	936	DEFW ADCAR7	5997788E8	1026	DEFW SKOKB
5979978E6	937	DEFW ADCAR7	599799FE5	1027	DEFW XRAt
5980178E6	938	DEFW ADCAR7	599819DE7	1028	DEFW PDDP
5980378E6	939	DEFW ADCAR7	599835AE7	1029	DEFW SELRB1
5980578E6	940	DEFW ADCAR7	5998588E6	1030	DEFW NIC
5980778E6	941	DEFW ADCAR7	5998772E4	1031	DEFW MP5WA
5980978E6	942	DEFW ADCAR7	599898FE5	1032	DEFW XRAR7
5981178E6	943	DEFW ADCAR7	599918FE5	1033	DEFW XRAR7
598138FE4	944	DEFW MXAa	599938FE5	1034	DEFW XRAR7
598158FE4	945	DEFW MXAa	599958FE5	1035	DEFW XRAR7
5981788E6	946	DEFW NIC	599978FE5	1036	DEFW XRAR7
59819AEE7	947	DEFW NAVR	599998FE5	1037	DEFW XRAR7
59821D4E7	948	DEFW SKOK	600018FE5	1038	DEFW XRAR7
59823A9E6	949	DEFW NAS0F0	600038FE5	1039	DEFW XRAR7
598257FE8	950	DEFW SKOKI	6000588E6	1040	DEFW NIC
5982788E6	951	DEFW NIC	6000788E6	1041	DEFW NIC
5982984E5	952	DEFW ORPt	6000988E6	1042	DEFW NIC
5983184E5	953	DEFW ORPt	600118AE4	1043	DEFW MP3AaA
5983384E5	954	DEFW ORPt	60013D4E7	1044	DEFW SKOK
5983588E6	955	DEFW NIC	6001542E7	1045	DEFW SELMB0
5983783E5	956	DEFW ODP47A	600170DE8	1046	DEFW SKOKC0
5983983E5	957	DEFW ODP47A	60019C6E5	1047	DEFW RLA
5984183E5	958	DEFW ODP47A	600210DE6	1048	DEFW DJNZRt
5984383E5	959	DEFW ODP47A	600230DE6	1049	DEFW DJNZRt
598459EE4	960	DEFW MXaA	600250DE6	1050	DEFW DJNZRt
598479EE4	961	DEFW MXaA	600270DE6	1051	DEFW DJNZRt
5984988E8	962	DEFW SKOKB	600290DE6	1052	DEFW DJNZRt
59851AEE7	963	DEFW NAVR	600310DE6	1053	DEFW DJNZRt
598539DE7	964	DEFW PDDP	600330DE6	1054	DEFW DJNZRt
59855CFE6	965	DEFW NAS0FN	600350DE6	1055	DEFW DJNZRt
5985723E8	966	DEFW SKOKZ1	6003741E4	1056	DEFW MAa
5985947E6	967	DEFW CARRY0	6003941E4	1057	DEFW MAa
598615EE5	968	DEFW ANLPt	6004188E8	1058	DEFW SKOKB
598635EE5	969	DEFW ANLPt	6004388E6	1059	DEFW NIC
598655EE5	970	DEFW ANLPt	600459DE7	1060	DEFW PDDP
5986788E6	971	DEFW NIC	600474AE7	1061	DEFW SELMB1
59869A9E5	972	DEFW ADP47A	6004902E8	1062	DEFW SKOKC1
59871A9E5	973	DEFW ADP47A	60051E1E5	1063	DEFW RLCA
59873A9E5	974	DEFW ADP47A	600532DE4	1064	DEFW MAR7
59875A9E5	975	DEFW ADP47A	600552DE4	1065	DEFW MAR7
5987747E4	976	DEFW MaA	600572DE4	1066	DEFW MAR7
5987947E4	977	DEFW MaA	600592DE4	1067	DEFW MAR7
5988188E6	978	DEFW NIC	600612DE4	1068	DEFW MAR7
59883AFE4	979	DEFW MPAA	600632DE4	1069	DEFW MAR7
59885D4E7	980	DEFW SKOK	600652DE4	1071	DEFW MAR7
59887B4E6	981	DEFW NAS1F0	600672DE4	1072	DEFW MAR7
5988988E6	982	DEFW NIC			
59891C6E6	983	DEFW CARRYN			

VÝPIS OBSAHU POLE Z\$

VÝPIS OBSLUŽNÉ ČÁSTI PROGRAMU PS-48

Z\$(1)=00	Z\$(65)=40	Z\$(129)=80	Z\$(193)=C0
Z\$(2)=01	Z\$(66)=41	Z\$(130)=81	Z\$(194)=C1
Z\$(3)=02	Z\$(67)=42	Z\$(131)=82	Z\$(195)=C2
Z\$(4)=03	Z\$(68)=43	Z\$(132)=83	Z\$(196)=C3
Z\$(5)=04	Z\$(69)=44	Z\$(133)=84	Z\$(197)=C4
Z\$(6)=05	Z\$(70)=45	Z\$(134)=85	Z\$(198)=C5
Z\$(7)=06	Z\$(71)=46	Z\$(135)=86	Z\$(199)=C6
Z\$(8)=07	Z\$(72)=47	Z\$(136)=87	Z\$(200)=C7
Z\$(9)=08	Z\$(73)=48	Z\$(137)=88	Z\$(201)=C8
Z\$(10)=09	Z\$(74)=49	Z\$(138)=89	Z\$(202)=C9
Z\$(11)=0A	Z\$(75)=4A	Z\$(139)=8A	Z\$(203)=CA
Z\$(12)=0B	Z\$(76)=4B	Z\$(140)=8B	Z\$(204)=CB
Z\$(13)=0C	Z\$(77)=4C	Z\$(141)=8C	Z\$(205)=CC
Z\$(14)=0D	Z\$(78)=4D	Z\$(142)=8D	Z\$(206)=CD
Z\$(15)=0E	Z\$(79)=4E	Z\$(143)=8E	Z\$(207)=CE
Z\$(16)=0F	Z\$(80)=4F	Z\$(144)=8F	Z\$(208)=CF
Z\$(17)=10	Z\$(81)=50	Z\$(145)=90	Z\$(209)=D0
Z\$(18)=11	Z\$(82)=51	Z\$(146)=91	Z\$(210)=D1
Z\$(19)=12	Z\$(83)=52	Z\$(147)=92	Z\$(211)=D2
Z\$(20)=13	Z\$(84)=53	Z\$(148)=93	Z\$(212)=D3
Z\$(21)=14	Z\$(85)=54	Z\$(149)=94	Z\$(213)=D4
Z\$(22)=15	Z\$(86)=55	Z\$(150)=95	Z\$(214)=D5
Z\$(23)=16	Z\$(87)=56	Z\$(151)=96	Z\$(215)=D6
Z\$(24)=17	Z\$(88)=57	Z\$(152)=97	Z\$(216)=D7
Z\$(25)=18	Z\$(89)=58	Z\$(153)=98	Z\$(217)=D8
Z\$(26)=19	Z\$(90)=59	Z\$(154)=99	Z\$(218)=D9
Z\$(27)=1A	Z\$(91)=5A	Z\$(155)=9A	Z\$(219)=DA
Z\$(28)=1B	Z\$(92)=5B	Z\$(156)=9B	Z\$(220)=DB
Z\$(29)=1C	Z\$(93)=5C	Z\$(157)=9C	Z\$(221)=DC
Z\$(30)=1D	Z\$(94)=5D	Z\$(158)=9D	Z\$(222)=DD
Z\$(31)=1E	Z\$(95)=5E	Z\$(159)=9E	Z\$(223)=DE
Z\$(32)=1F	Z\$(96)=5F	Z\$(160)=9F	Z\$(224)=DF
Z\$(33)=20	Z\$(97)=60	Z\$(161)=A0	Z\$(225)=E0
Z\$(34)=21	Z\$(98)=61	Z\$(162)=A1	Z\$(226)=E1
Z\$(35)=22	Z\$(99)=62	Z\$(163)=A2	Z\$(227)=E2
Z\$(36)=23	Z\$(100)=63	Z\$(164)=A3	Z\$(228)=E3
Z\$(37)=24	Z\$(101)=64	Z\$(165)=A4	Z\$(229)=E4
Z\$(38)=25	Z\$(102)=65	Z\$(166)=A5	Z\$(230)=E5
Z\$(39)=26	Z\$(103)=66	Z\$(167)=A6	Z\$(231)=E6
Z\$(40)=27	Z\$(104)=67	Z\$(168)=A7	Z\$(232)=E7
Z\$(41)=28	Z\$(105)=68	Z\$(169)=A8	Z\$(233)=E8
Z\$(42)=29	Z\$(106)=69	Z\$(170)=A9	Z\$(234)=E9
Z\$(43)=2A	Z\$(107)=6A	Z\$(171)=AA	Z\$(235)=EA
Z\$(44)=2B	Z\$(108)=6B	Z\$(172)=AB	Z\$(236)=EB
Z\$(45)=2C	Z\$(109)=6C	Z\$(173)=AC	Z\$(237)=EC
Z\$(46)=2D	Z\$(110)=6D	Z\$(174)=AD	Z\$(238)=ED
Z\$(47)=2E	Z\$(111)=6E	Z\$(175)=AE	Z\$(239)=EE
Z\$(48)=2F	Z\$(112)=6F	Z\$(176)=AF	Z\$(240)=EF
Z\$(49)=30	Z\$(113)=70	Z\$(177)=B0	Z\$(241)=F0
Z\$(50)=31	Z\$(114)=71	Z\$(178)=B1	Z\$(242)=F1
Z\$(51)=32	Z\$(115)=72	Z\$(179)=B2	Z\$(243)=F2
Z\$(52)=33	Z\$(116)=73	Z\$(180)=B3	Z\$(244)=F3
Z\$(53)=34	Z\$(117)=74	Z\$(181)=B4	Z\$(245)=F4
Z\$(54)=35	Z\$(118)=75	Z\$(182)=B5	Z\$(246)=F5
Z\$(55)=36	Z\$(119)=76	Z\$(183)=B6	Z\$(247)=F6
Z\$(56)=37	Z\$(120)=77	Z\$(184)=B7	Z\$(248)=F7
Z\$(57)=38	Z\$(121)=78	Z\$(185)=B8	Z\$(249)=F8
Z\$(58)=39	Z\$(122)=79	Z\$(186)=B9	Z\$(250)=F9
Z\$(59)=3A	Z\$(123)=7A	Z\$(187)=BA	Z\$(251)=FA
Z\$(60)=3B	Z\$(124)=7B	Z\$(188)=BB	Z\$(252)=FB
Z\$(61)=3C	Z\$(125)=7C	Z\$(189)=BC	Z\$(253)=FC
Z\$(62)=3D	Z\$(126)=7D	Z\$(190)=BD	Z\$(254)=FD
Z\$(63)=3E	Z\$(127)=7E	Z\$(191)=BE	Z\$(255)=FE
Z\$(64)=3F	Z\$(128)=7F	Z\$(192)=BF	Z\$(256)=FF

```

1 GO TO 1000
5 REM
10 REM vstupni parametry
20 LET adresa=57936
21 LET dis=adresa+35
22 LET pc48=adresa+25
23 LET simul=adresa+38
24 LET byte=adresa+30
25 LET hodiny=adresa+32
26 LET aku=adresa+22
27 LET frek=3
28 LET psw=adresa+23
29 LET time=adresa+24
30 LET ztt=61000
31 LET int=adresa+14
32 LET t0=adresa+12
33 LET t1=adresa+13
34 LET port0=adresa+15
35 LET konadr=adresa+27
36 LET rysi=adresa+41
40 LET zpp=53248
50 LET tz=57856
60 LET zramp=57344
65 LET zram=zramp+256: R

```

```

ETURN
66 LET kq=1: DIM z$(256,
2): FOR i=48 TO 70: IF i=58 THEN
LET i=65
67 FOR j=48 TO 70: IF j=
58 THEN LET j=65
68 LET z$(kq)=CHR$ i+CHR
$ j: LET kq=kq+1: NEXT j: NEXT i
RETURN
75 PRINT AT 18,0;"

```

```

"; RETURN
100 REM POKE na 2 adresy
105 LET d1=INT (c/256): L
ET d2=c-d1*256: POKE adr,d2: POK
E

```

```

adr+1,d1: RETURN
1000 REM HLAVNI PROGRAM

```

```

1001 PAPER 4: INK 0: BORDE
R 4: CLS : DIM k$(3,3): DIM z(3)
:

```

```

LET p=0: LET k=1: LET k$(1
)="h": LET k$(2)="d": LET
k$(3)="b": LET z(1)=16: LE
T z(2)=10: LET z(3)=2
1005 GO SUB 10
1011 DEF FN a(t)=PEEK t+25
6*PEEK (t+1)
1115 GO SUB 1994
1120 POKE 23658,8: PAUSE 0
: BEEP .01,20: LET y$=INKEY$
1122 IF y$="D" THEN GO SU
B 2050: GO SUB 1994
1124 IF y$="I" THEN GO SU
B 2010
1125 IF y$="AT " THEN GO
SUB 3800: GO SUB 75
1126 IF y$="Q" THEN GO SU
B 3850: GO SUB 75
1127 IF y$="H" THEN GO SU
B 3820: GO SUB 75
1128 IF y$="R" THEN GO SU
B 2030: GO SUB 75
1130 IF y$="P" THEN GO SU
B 2040: GO SUB 75
1131 IF y$="K" THEN GO SU
B 3840: GO SUB 75
1132 IF y$="W" THEN GO SU
B 3600: GO SUB 1994

```

```

1133 IF y$="B" THEN GO SU
B 3870
1134 IF y$="L" THEN GO SU
B 3700: GO SUB 1994
1135 IF y$="T" THEN RANDO
MIZE USR 57980: GO SUB 1995
1136 IF y$="Z" THEN LET h
mez=64: LET rozsah=8: LET typ=zr
am
: CLS : GO SUB 3200: GO SU
B 1994
1138 IF y$="X" THEN LET h
mez=256: LET rozsah=16: LET typ=
zr
amp: CLS : GO SUB 3200: GO
SUB 1994
1140 IF y$="C" THEN LET h
mez=4096: LET rozsah=20: LET typ
=z
pp: CLS : GO SUB 3200: GO
SUB 1994
1142 IF y$="1" THEN PRINT
AT 18,0;"A": LET adr=aku: GO SU
B
2060: PRINT AT 11,4;c$
1144 IF y$="2" THEN PRINT
AT 18,0;"PSW": LET adr=psw: GO
SU
B 2060: PRINT AT 11,14;c$
1146 IF y$="3" THEN PRINT
AT 18,0;"T": LET adr=time: GO S
UB
2060: PRINT AT 11,22;c$
1148 IF y$="0" THEN GO SU
B 2070
1150 IF y$=":" THEN LET t
yp=zram: LET c$="Z": LET hmez=64
:
GO SUB 3300: GO SUB 1994
1152 IF y$=";" THEN LET t
yp=zramp: LET c$="X": LET hmez=1
28
: GO SUB 3300: GO SUB 1994
1154 IF y$="?" THEN LET t
yp=zpp: LET c$="C": LET hmez=409
6:
GO SUB 3300: GO SUB 1994
1156 IF y$="M" THEN GO SU
B 3880: GO SUB 1994
1990 GO TO 1120
1993 REM Panel
1994 CLS : GO SUB 2020
1995 PRINT AT 0,2: GO SUB
2000: GO SUB 2026: RETURN
1999 REM DISASSEMBLER
2000 RANDOMIZE USR dis: P
RINT TAB (2);
2005 FOR i=tz TO tz+20: PR
INT CHR$ (PEEK i);: NEXT i: RETU
RN
2009 REM INTERPRET
2010 RANDOMIZE USR simul:
GO SUB 2026: PRINT AT 0,2: GO S
UB
2000: RETURN
2019 REM REGISTRY
2020 PRINT AT 2,0: FOR i=
0 TO 7: PRINT "R";i;: -": NEXT
i:
PRINT "A - ";: PRINT TAB
8;"PSW - ";: PRINT TAB 18;"T -
"
2021 PRINT AT 2,20;"BUS-":
FOR i=1 TO 7: IF i=3 THEN NEX
T
i
2025 PRINT AT 2+i,20;"P";i
;: -": NEXT i: RETURN
2026 LET kq=0: FOR i=zram
TO zram+7: PRINT AT 2+kq,5;z$(PE
EK
i+1);: PRINT " ";z$(PEE
K (i+24)+1): LET kq=kq+1: NEXT i
:
PRINT AT 11,4;z$(PEEK aku+
1);: PRINT AT 11,14;z$(PEEK psw+
1)
;: PRINT AT 11,22;z$(PEEK
time+1)

```

```

2027 LET a=port0: FOR i=0
TO 7: IF i=3 THEN NEXT i
2028 PRINT AT 2+i,25;z$(PE
EK a+1): LET a=a+1: NEXT i: RETU
RN
2029 REM POKE na registr
2030 INPUT "sada 0/1 ";sad
a: IF sada<>0 AND sada<>1 THEN
GO
TO 2030
2031 PRINT AT 18,3;"sada "
;sada
2033 INPUT "registr 0..7 "
;registr: IF registr<0 OR regist
r>
7 THEN GO TO 2033
2034 PRINT "registr ";regi
str
2035 PAUSE 20: PRINT AT 20
,2;"obsah ";: GO SUB 3045: GO SU
B
3010: LET c=c-INT (c/256)*
256: IF p=0 THEN POKE (zram+reg
is
tr+sada*24),c: GO SUB 3030
: PRINT AT 2+registr,5+sada*5;c$
:
RETURN
2038 GO SUB 3005: GO TO 20
35
2039 REM POKE na porty
2040 INPUT "port ";port: I
F port<0 OR port>7 OR port=3 THE
N
GO TO 2040
2041 PRINT AT 18,3;"port "
;port
2042 PAUSE 20: PRINT AT 20
,2;"obsah ";: GO SUB 3045: GO SU
B
3010: LET c=c-INT (c/256)*
256: IF p=1 THEN GO SUB 3005: G
O
TO 2042
2044 IF port>2 THEN LET p
ort=port-1: LET c=c-INT (c/16)*1
6
2046 POKE port0+port,c: IF
port>2 THEN LET port=port+1
2048 PRINT AT 2+port,25;z$
(c+1): RETURN
2049 REM Vypis programu
2051 CLS : LET port=FN a(p
c48)
2052 POKE 23692,255
2054 FOR j=0 TO 10: GO SUB
2000: PRINT : LET c=PEEK (byte)
+F
N a(pc48): LET adr=pc48: G
O SUB 100: NEXT j: PAUSE 0: IF I
NK
EY$<>" " THEN GO TO 2052
2056 LET c=port: LET adr=p
c48: GO SUB 100
2058 RETURN
2059 REM Akumulator PSW Ti
mer
2060 PRINT AT 18,5;"obsah
";: PAUSE 20: GO SUB 3045: GO SU
B
3010: IF p=0 THEN LET c=c
-INT (c/256)*256: POKE adr,c: GO
S
UB 75: GO SUB 3030: RETURN
2062 GO SUB 3005: GO TO 20
60
2069 REM Definovani PC48
2070 PRINT AT 18,0;"PC48"
2071 PRINT AT 18,5;"obsah
";: PAUSE 20: GO SUB 3045: GO SU
B
3010: IF p=0 THEN LET c=c
-INT (c/4096)*4096: LET adr=pc48
:
GO SUB 100: GO SUB 75: PRI
NT AT 0,2: GO SUB 2000: RETURN
2072 GO SUB 3005: GO TO 20
71
3000 REM Chyba

```

```

3005 BEEP .1,0: BEEP .1,15
: BEEP .1,30: PAUSE 30: BEEP .1,
0:
BEEP .1,15: BEEP .1,30: R
ETURN
3009 REM Prevod c(z)->c(10
)
3010 LET c=0: LET d=LEN a$
: FOR i=1 TO d: LET c$=a$(i TO i
):
GO SUB 3020: IF c1<z(k) A
ND c1>=0 THEN LET c=c+c1*(z(k)^
(d
-i)): NEXT i: LET p=0: RET
URN
3015 LET p=1: RETURN
3020 LET c1=CODE c$: IF c1
>57 THEN LET c1=c1-7
3025 LET c1=c1-48: RETURN
3029 REM Prevod c(10)->c(1
6)
3030 LET c$=z$(c+1): RETUR
N
3044 REM Nacitani pres INK
EY$
3045 LET p=0: LET c$=""
3050 PRINT k$(k);CHR$ 8;:
LET a$=INKEY$: IF a$="" THEN G
O
TO 3050
3051 BEEP .05,20: IF a$=""
" THEN PRINT a$: RETURN
3052 IF CODE a$=10 OR CODE
a$=11 THEN GO SUB 3090: GO TO
30
50
3053 IF CODE a$=13 THEN P
RINT " ": LET a$=c$: RETURN
3054 IF CODE a$=12 THEN G
O TO 3060
3056 LET p=1: LET c$=c$+a$
: PRINT a$: GO TO 3050
3060 LET c=LEN c$: IF c>0
THEN PRINT " ";CHR$ 8;CHR$ 8;:
LE
T c$=c$( TO (c-1)): GO TO
3050
3061 LET p=0: GO TO 3050
3090 IF p=0 THEN LET k=k+
1: IF k>3 THEN LET k=1: RETURN
3091 RETURN
3199 REM Listing pameti
3200 PRINT "D";y$;: ";: PO
KE 23658,8: GO SUB 3045: IF a$=""
"
THEN RETURN
3205 GO SUB 3010: IF p=1 T
HEN GO SUB 3005: GO TO 3200
3206 LET a=c: GO SUB 3208:
GO TO 3210
3208 IF a>hmez THEN LET
a=a-hmez: GO TO 3208
3209 RETURN
3210 POKE 23692,255: PRINT
TAB 0;: FOR i=1 TO rozsah: GO S
UB
3208: LET c=INT (a/256):
PRINT " ";z$(c+1);: LET c=a-INT
(a
/256)*256: PRINT z$(c+1);"
";: FOR j=0 TO 7
3219 IF INKEY$="" " THEN R
ETURN
3220 GO SUB 3208: PRINT "
";z$(PEEK (a +typ)+1);: LET a=a+
1:
NEXT j: PRINT : NEXT i
3226 PAUSE 20: PAUSE 0: IF
INKEY$="" " THEN RETURN
3230 GO TO 3210
3299 REM Ukladani do pameti
i
3300 CLS : POKE 23658,8: P
RINT "S";c$;: ";: GO SUB 3045: I
F
a$="" " THEN RETURN
3305 GO SUB 3010: IF p=1 T
HEN GO SUB 3005: GO TO 3300
3306 LET a=c
3307 POKE 23692,255: GO SU
B 3208

```

```

3310 LET c=INT (a/256): PR
INT " ";z$(c+1);: LET c=a-INT (a
/2
56)*256: PRINT z$(c+1);"
";: PRINT " ";z$(PEEK (typ+a)+1)
;
- ";: GO SUB 3045: IF a$=
" " THEN RETURN
3311 IF c$="" THEN GO TO
3316
3312 GO SUB 3010: IF p=1 T
HEN BEEP .5,-20: GO TO 3307
3314 IF c>255 THEN LET c=
c-INT (c/256)*256
3315 POKE a+typ,c
3316 LET a=a+1: GO TO 3307
3600 CLS : POKE 23692,255:
PRINT AT 1,10;"Zápis programu";
:
INPUT "jméno";j$: IF j$=""
THEN RETURN
3601 POKE 23658,8
3602 PRINT AT 2,10;"jméno
":j$: BEEP .1,20
3605 PRINT AT 5,10;"od - "
;: GO SUB 3045: IF a$="" " OR c$=
""
THEN RETURN
3610 GO SUB 3010: IF p=1 T
HEN GO SUB 3005: GO TO 3605
3611 LET a=c: GO SUB 3208
3615 PRINT AT 10,10;"do -
";: GO SUB 3045: IF a$="" " OR c$=
""
THEN RETURN
3620 GO SUB 3010: IF p=1 T
HEN GO SUB 3005: GO TO 3615
3625 LET c=c-INT (c/65536)
*65536: IF c-a<=0 THEN GO SUB 3
00
5: GO TO 3605
3630 SAVE j$CODE (zpp+a),c
-a: GO SUB 5: PRINT AT 15,3;"Chc
eš
provést VERIFY - a/n": P
AUSE 0: IF INKEY$="A" THEN VERI
FY
j$CODE
3635 RETURN
3699 REM Nahravani program
u
3700 CLS : PRINT AT 8,8;"N
atažení programu";: INPUT "jméno
";
j$: IF j$="" THEN RETURN
3710 PRINT AT 15,10;"jméno
":j$: LOAD j$CODE : GO SUB 30
05
: RETURN
3799 REM Bloková simulace
3800 PRINT AT 18,0;"koncov
á adresa ";: PAUSE 20: GO SUB 30
45
: GO SUB 3010: IF p=1 THEN
GO SUB 3005: GO TO 3800
3801 PAUSE 10: LET port=c:
PRINT AT 19,0;"krokování a/n
";
: PAUSE 0: LET c$=INKEY$:
IF c$="n" OR c$="N" THEN LET c$=
""
n": GO TO 3803
3802 LET c$="a"
3803 PRINT c$
3804 RANDOMIZE USR dis: PR
INT AT 20,0: FOR i=tz+1 TO tz+3:
P
RINT CHR$ (PEEK i);: NEXT
i: IF c$="a" THEN PAUSE 0: IF I
NK
EY$="" THEN GO TO 3810
3805 IF INKEY$="" THEN G
O TO 3810
3806 RANDOMIZE USR dis: RA
NDOMIZE USR simul: IF FN a(pc48)
<>
port THEN GO TO 3804
3810 GO SUB 1995: RETURN
3819 REM Hodiny
3820 LET c=PEEK hodiny+256
*PEEK (hodiny+1)+65536*PEEK (hod
in

```

```

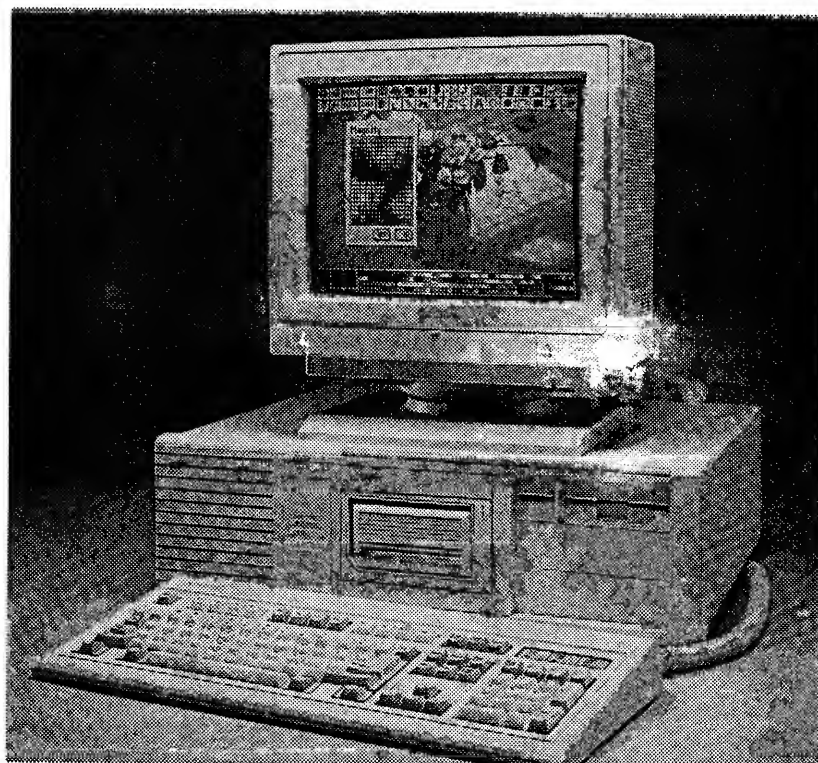
y+2)
3825 PRINT AT 18,0;"kmitočet
et krystalu = ";fрек;" MHz": PRI
NT
AT 20,0;3/fрек*5*c/1000;"
[ms]": PAUSE 0: IF INKEY$=""0" T
HE
N POKE hodiny,0: POKE hod
iny+1,0: POKE hodiny+2,0
3830 RETURN
3839 REM Kmitočet krystalu
3840 PRINT AT 20,0;"kmitočet
et krystalu = ";fрек;" MHz"
3845 INPUT "kmitočet [MHz]
=";fрек: RETURN
3849 REM T0 - T1 - INT
3850 PRINT AT 18,0;"INT -
";PEEK int;TAB 10;"T0 - ";PEEK t
0;
TAB 20;"T1 - ";PEEK t1
3852 PAUSE 0: LET c$=INKEY
$: IF c$="" THEN RETURN
3853 IF c$="1" THEN LET t
yp=int: GO TO 3857
3854 IF c$="2" THEN LET t
yp=t0: GO TO 3857
3855 IF c$="3" THEN LET t
yp=t1: GO TO 3857
3856 GO TO 3852
3857 IF PEEK typ=1 THEN P
OKE typ,0: GO TO 3850
3858 POKE typ,1: GO TO 385
0
3859 REM Rychlá simulace
3870 PRINT AT 18,0;"rychlá
simulace";AT 19,0;"koncová adre
sa
";: PAUSE 20: GO SUB 3045
: GO SUB 3010: IF a$="" " OR c$=
""
THEN RETURN
3872 IF p=1 THEN GO TO 38
70
3873 IF a$="" THEN GO SU
B 75: RETURN
3875 POKE konadr+1,INT (c/
256): POKE konadr,(c-INT (c/256)
*2
56): RANDOMIZE USR rys: G
O SUB 75: GO SUB 1995: RETURN
3879 REM Obsluha programu
3880 CLS : POKE 23692,255:
PRINT " Obsluha PS - 48

```

```

3890 PRINT BRIGHT 1;" de
finování/listování paměti
#C/C - programu
#Z/Z - dat
#X/X - dat (vnější)
3895 PRINT " de
finování R - registr
u
P - portu 0 - PC4
8 1 - A 2 - PSW
3 - T"
3900 PRINT BRIGHT 1;" D
- disassembler I
-
kroková simulace
#I - bloková simulace
B - bloková simulace II.
3910 PRINT " Q - jednobit
ové vstupy 1 - nega
ce
INT 2 -
negace T0 3 -
ne
gace T1
3920 PRINT BRIGHT 1;" W
- záznam paměti programu L
-
definování paměti program
u"
3930 PRINT " H - stav ho
din 0 - nul
ov
ání K - kmi
točet oscilátoru ";#0; BRI
GHT
1;" ^ - změna číselné so
ustavy ";
3940 PAUSE 0
3945 RETURN
9996 BEEP .5,10: BEEP .1,3
0: PRINT #0; PAPER 4; INK 0;AT 1
,0;
" program je nahrán
stiskněte klávesu
"
9998 PAUSE 0: GO TO 1000

```



EMULÁTOR

PRE JEDNOČIPOVÉ MIKROPOČÍTAČE NA ZX-SPECTRU

Ing. Ján Sokol, Hranická 15, 750 00 Píerov

S rozvojom výpočtovej techniky sa stále viac používajú mikroprocesory na riadenie rôznych elektronických prístrojov. Táto tendencia sa prejavuje aj u nás, o čom svedčí zahájenie výroby jednočipových mikropočítačov v k.p. TESLA Plešňany. Možnosti týchto malých riadiacich mikropočítačov sú vskutku veľké. Nasledovný príspevok má za cieľ prispieť k rozšíreniu používania jednočipových mikropočítačov aj v amatérskych podmienkach. Jedná sa o malý vývojový systém a emulátor pre jednočipové mikropočítače 8035-8048, riešený ako doplnok k mikropočítaču Sinclair ZX-Spectrum.

Emulátor ZX-48 je určený pre tvorbu a odlaďovanie programového vybavenia na jednočipové mikropočítače 8035/48. Základnou požiadavkou pri jeho stavbe bola konštrukčná jednoduchosť, použiteľnosť dostupných súčiastok a prenesenie ťažiska ovládacích funkcií v maximálnej miere na programové vybavenie. Emulátor je teda plne ovládaný z riadlaceho mikropočítača ZX-Spectrum. Odladovaný program je uložený v pamäti ZX-Spectrum, odkiaľ je po jednotlivých inštrukciách predávaný mikropočítaču na doske emulátora.

Rozsiahle programové vybavenie na ZX-Spectrum umožňuje tvorbu programov (pomocou riadkového prekladača), ich opravy a overenie ich správnej funkcie. Program je možné krokovat, alebo spustiť s vopred definovanými bodmi zastavenia či bez zastavenia. Na obrazovke sú v základnom režime zobrazené aktuálne obsahy registrov výkonného mikropočítača 8035 resp. 8048. Jednoduchým spôsobom je možné si prehliadnúť aktuálny obsah vnútornej datovej pamäte, prípadne zmeniť obsah ktorejkoľvek buňky či registra. Takisto je možné si prehliadnúť pamäť programu v tvare strojového kódu vyjadreného hexadecimálnymi znakmi, alebo v tvare dekompilovaných inštrukcií. Program je možné opravovať buď priamo zmenou obsahu príslušných pamäťových buniek, alebo zmenou textu príslušnej inštrukcie.

Emulátor tiež umožňuje prepísať obsah pamäte programu z pripojenej aplikácie (príp. obsah pamäte EPROM na čípe 8748) do pracovnej oblasti pamäte programu v RAM ZX-Spectrum. Emulátor ZX-48 má ešte jednu dôležitú vlastnosť, ktorú užívateľ ocení hlavne pri odladovaní programov, v ktorých je pre komunikáciu s okolím využívaný adresovateľný priestor vonkajšej datovej

pamäte. V pamäti ZX-Spectrum je možné totiž vopred definovať obsah vonkajšej datovej pamäte, na ktorú sa potom výkonný procesor obracia pri inštrukciách MOVX A. Kedykoľvek je možné si obsah tejto pamäte prehliadnúť, príp. jej obsah zmeniť alebo ju odpojiť a výkonný procesor potom pri inštrukciách MOVX A číta údaje z reálnej aplikácie, na ktorú je pripojený cez emulačný kábel.

Popis programu

Program pre riadenie emulátora je složený zo dvoch častí. Časť programu napísaná v BASICu zabezpečuje riadenie programu, ovládanie systému menu, inicializáciu premenných ako aj inicializáciu hardware. Z BASICu sa pre vykonanie väčšiny funkcií volajú podprogramy v strojovom kóde. Adresy vstupných bodov týchto podprogramov sa načítavajú do premenných. Celý program je riadený systémom niekoľkých menu. Požadovaný príkaz (alebo prechod do ďalšieho podmenu) sa realizuje stlačením jednej z kláves 1, 2, 3, 8, 9, 0. Platné funkcie týchto kláves sú neustále zobrazené v tabuľke na spodnom okraji obrazovky. Celá štruktúra menu a prechody z jedného menu do druhého sú znázornené v prílohe č.3. Ovládanie systému menu je riadené pomocou poľí L(,,) a H\$(,). Pole L(,,) obsahuje čísla riadkov programu v BASICu, na ktorých začína obsluha jednotlivých funkcií, pričom prvý index určuje číslo menu a druhý index určuje číslo funkcie v tomto menu.

Ihneď po nahraní programu sa na obrazovke objaví „základné okno“, v ktorom sú zobrazené obsahy registrov jednočipového mikropočítača na doske emulátora (ďalej nazývaného

Systém menu emulátora

MAIN MENU:

- 1 copy
- 2 reset
- 3 main mode
- 8 edit
- 9 list
- 0 step

COPY:

- 1 8035
- 2 8048
- 8 move
- 9 fill
- 0 exit

MAIN MODE:

- 1 mode real
- 2 mode simulace
- 3 real
- 8 run
- 9 break
- 0 step

EDIT:

- 1 externí data memory
 - 2 interní data memory
 - 3 break point memory
 - 8 program
 - 9 program memory
 - 0 tape
- (obsahuje ďalšie 3 submenu)

LIST:

- 1 ext. data memory
- 2 int. data memory
- 3 break point memory
- 8 program
- 9 program memory
- 0 exit



výkonný procesor), a vykonávaná inštrukcia. V hornej časti obrazovky je v tabuľke zobrazený obsah registrov a stavové slovo (PSW) procesora. Nad touto tabuľkou je zobrazená inštrukcia, ktorá sa má v nasledujúcom kroku vykonať. Stavové slovo (PSW) je vyjadrené v hexadecimálnej forme i v binárnej forme, pričom nad každým bitom je v hlavičke tabuľky nadpísaný význam informácie tohto bitu.

Strojové podprogramy

Strojová časť ovládacieho programu začína od adresy 49000 (BF68h) a skladá sa z niekoľkých častí.

1. Tabuľka inštrukčných kódov 8035/48 (BF68 - C04E).
2. Tabuľka textov inštrukcií 8035/48 (C04E - C953).
3. Kópia vnútornej pamäte dát (C96D - C9AC).
4. Náhradná vonkajšia pamäť dát (C9AD - CAAC).
5. Podprogramy volané z BASICu (CAAD - D280).
6. Inštrukčné kódy 8035 pre realizovanie niektorých funkcií (reset, zistenie obsahu registrov ap.) (D281 - D2DD).
7. Obsah video-pamäte (D2DC - EA3B).
8. Oblasť pamäte programu (EE46 - FE45).
9. Oblasť pre záznam bodov zastavenia (od FE46).

Popis zapojenia

Schéma zapojenia emulátora je na obr.1. Emulátor sa pripája na zbernicu ZX Spectrum, pričom napájanie je riešené samostatne cez stabilizátor napätia U7, takže vnútorný stabilizátor v Spectre nie je zafarazený. ZX Spectrum komunikuje s procesorom 8035 pomocou obvodu 8255 a troch osemblietových registrov U3,U4,U5 na nasledovných adresách:

port A :	31
port B :	63
port C :	95
riad. register :	127
register R1 :	159
register R2 :	191
register R3 :	223

Diódy D1 - D4 zaisťujú aktívnu úroveň signálu CS pre obvod 8255 pri vstupných/výstupných operáciách na prvých štyroch vyššie uvedených adresách. Obvod 8255 pracuje v kombinovanom móde. Skupina A (t.j. port A a nižšie bity portu C) pracuje ako vstup v móde 0 a skupina B (t.j. port B a vyš-

šie bity portu C) pracuje ako vstup v móde 1. Do portu B sa zapíše slovo z datovej zbernice procesora 8035 vždy pri inštrukciách zápisu do vonkajšej pamäte dát (t.j. pri inštrukciách MOVX @R0,A a MOVX @R1,A).

Obvody U3,U4 pracujú ako výstupné registre. Výstup ich obsahu na datovú zbernicu 8035 sa riadi logickými signálmi z riadiacej logiky. Obvod U5 pracuje ako výstupný register, ktorého výstup je stále uvoľnený (vývod OE je uzemnený) a v zapojení má riadiacu funkciu. Jednotlivé bity tohto registra ovládajú funkciu emulátora v rozličných módoch.

Riadiaca logika na strane procesora 8035 má nasledujúce funkcie:

Klopny obvod U8A riadi krokovanie programu. Je nastavený vždy zápisom do registra R1. Tým sa signál SS dostane do úrovne H a procesor 8035 vykonáva 1 inštrukciu, pretože klopny obvod je vzápätí resetovaný akonáhle začne čítanie 1. bajtu inštrukcie (impulz úrovne L na signále ALE).

Klopny obvod U8B rozlišuje, ktorý bajt inštrukcie sa práve číta. Je resetovaný vždy po zápise do R2, a preklopí sa pri aktívnom signále RD alebo PSEN. Celá činnosť emulátora je riadená pomocou signálov registra R3 nasledovným spôsobom:

Nultý bit (E0) riadi prenos údajov pri inštrukciách čítania z vonkajšej pamäte dát (MOVX A,@R0 a MOVX A,@R1). Ak je v úrovni L, tak signál RD z procesora 8035 sa cez hradia U10A a U12B dostáva do aplikácie a procesor číta reálne údaje. Naopak, ak je signál E0 v úrovni H, tak signál RD v aplikácii je trvale v úrovni H a čítanie sa uskutoční z registra R2. Obslužný program v ZX Spectrum vopred do tohto registra zapíše vhodný údaj z oblasti 'fiktívnej vonkajšej pamäte'. Druhý bit (E2) v úrovni L udržiava klopny obvod U8A trvale nastavený a teda signál SSTEP je stále v úrovni H a program beží plnou rýchlosťou. Ak je signál E2 v úrovni H, klopny obvod sa po každej inštrukcii resetuje (nastavuje sa pri zápise do registra R1) a program sa vykonáva po krokoch. Tretí bit (E3) pri úrovni L drží klopny obvod U9B trvale nastavený a tým zamedzuje čítanie 1. bajtu z registra R1. Platí bit (E5) určuje, odkiaľ procesor 8035 prečíta pri dvobajtových inštrukciách druhý bajt inštrukcie. Pri úrovni L sa číta 2. bajt z aplikácie (signál PSEN je uvoľnený cez hradlo U12D) a pri úrovni H sa číta 2.bajt z registra R2 (signál PSEN je uvoľnený cez hradlo U12C). Siedmy bit (E7) udržiava úroveň H na vstupe EA procesora. Tento vstupný signál je využitý pri procesoroch 8048 a 8748. Pri úrovni H pracujú tieto procesory s vonkajšou pamäťou programu, ako 8035. Ak je jeho úroveň L, inštrukcie sa čítajú z ROM (resp EPROM) na čipe. Ak teda tento signál prejde z úrovne H do úrov-

ne L tesne pred čítaním 2. bajtu, tak je možné takto prečítať obsah ROM (resp. EPROM) na čipe procesora 8048, resp. 8748.

Vhodným kombinovaním uvedených signálov je ZX Spectrum schopné zabezpečiť všetky potrebné funkcie na ovládanie emulátora.

Záver

S popisovaným emulátorom, postaveným na univerzálnej doske, som vyvinul a odladil niekoľko aplikácií systémov s jednočipovým mikropočítačom 8035. Pri rozsiahlejších programoch sa prejavuje citeľne potreba úplného prekladača assembleru 8035/48. Na toto už 'malé' ZX Spectrum nemôže stačiť hlavne z dôvodov kapacity pamäte pre zdrojový text programu. Tento problém sa dá riešiť napr. prekladom programu na niektorom vývojovom systéme (napr. SAPI 80) a prenosom preloženého kódu do ZX Spectra (sériová linka pre Spectrum dnes už nie je problém) a jeho následným ladením pomocou uvedeného emulátora, podobne ako je to pri používaní kufrikového emulátora TEMS-49. V tomto prípade má popisovaný emulátor dokonca oproti TEMS-49 ešte veľkú výhodu v komforte obsluhy.

Toto zariadenie však nie je určené pre rozsiahle programy, rovnako ako jednočipové mikropočítače nie sú určené pre riešenie rozsiahlych úloh. Pre väčšinu bežných aplikácií s jednočipovými mikropočítačmi však postačuje a môže sa stať užitočným pomocníkom ak jednotlivcom, tak i v záujmových kružkoch.

Záujemcom o stavbu emulátora som ochotný nahrnúť na kazetu kompletný ovládaci program a tiež text s podrobným návodom na ovládanie.

(Vzhľadom k dĺžke výpisu programu nebylo možné jej uverejniť - pozn. red.)

Literatúra

- [1] Jednočipové mikropočítače 8048.
- [2] Interfejs k ZX-Spectru s 8255. AR 6/85.
- [3] Šály, M.: Porty k mikropočítači. AR 8/85.
- [4] Meduna, S.: Vstupné a výstupné porty. AR 10/85.
- [5] Vickers, S.: Sinclair ZX-Spectrum Basic programming.
- [6] Horsák, I.: Jednoduchý prevod parametrov z BASICu do strojového kódu u Spectra. AR 3/87.

LOGICKÝ ANALYZÁTOR

Juraj Šrámek, Asmolvova 53, 842 47 Bratislava

Pomocou popísaného zariadenia je možné využiť počítač ATARI XL/XE na rôzne merania v číslicových systémoch. Logický analyzátor umožňuje merať naraz ôsmich číslicových signálov v dvoch režimoch. Pri jeho stavbe som vychádzal zo zapojenia Ing. Dovala, uverejneného v AR A 2,3/89. Toto zapojenie som upravil tak, aby bolo možné všetky funkcie ovládať programovo z nadradeného počítača.

Logický analyzátor umožňuje pracovať vo dvoch režimoch:

- asynchrónnom (signál sa vzorkuje vnútornými hodinovými impulzmi),
- synchronnom (vzorkovací signál je privedený na vstup F in).

Spustenie vzorkovania môže byť

- interné (odvodené od vstupného slova),
- externé (vstup EXT).

Zapojenie pozostáva z dvoch častí:

- paralelný port,
- logický analyzátor.

Doska portov (obr. 1)

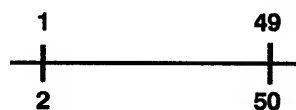
Doska portov je osadená dvomi obvodmi MHB8255 (PPI), kryštálovým oscilátorom s deličmi a dekodérom riadiacich signálov a adres mikroprocesora 6502.

Vstupno-výstupné linky obvodu PPI, adresové vstupy a vstup D7 multiplexera U9, a výstupy signálov sú vyvedené na konektor. Sem sa môžu pripojiť rôzne periférie zariadenia (tlačiareň, plotter, atď...), čiže aj doska logického analyzátoru.

Hradlá U5A, U4A a U4B slúžia ako dekodér adres, hradlá U3A a U3B vyrábajú zo signálov R/W a O2 μ P 6502 signály RD a WR. Invertor U5B mení úroveň signálu RST pre obvod PPI.

Doska portov musí byť pripojená na zbernicu počítača, ale tá je pri typoch XE a XL vyvedená odlišným spôsobom. Typ XL ju má vyvedenú na jednom konektore „Parallel Bus“, ale nie je na ňom napájanie ani signál D7xx. Obidva je potrebné príviesť na voľné plošky konektora (signál D7xx privedieme z pinu číslo 7 obvodu 74LS138, ktorý sa nachádza v strede plošného spoja pri konektore pre cartridge). Zapojenie konektora zbernice je na obr. 2. Typ XE má vyvedený signál D5xx, ktorý sa aktivuje pri adresovaní oblasti \$D500 až \$D5FF, preto je nutné zmeniť príslušné adresy portov v programe.

GND	1	2	EXTSEL
A0	3	4	A1
A2	5	6	A3
A4	7	8	A5
A6	9	10	GND
A7	11	12	A8
A9	13	14	A10
A11	15	16	A12
A13	17	18	A14
GND	19	20	A15
D0	21	22	D1
D2	23	24	D3
D4	25	26	D5
D6	27	28	D7
GND	29	30	GND
2 (B02)	31	32	GND
D7xx	33	34	RST
IRQ	35	36	RDY
n.c.	37	38	EXTENB
n.c.	39	40	REFRESH
CAS	41	42	GND
MPD	43	44	RAS
GND	45	46	R/W (LR/W)
+5V	47	48	n.c.
+5V	49	50	GND



Obr. 2. Zapojenie konektora Parallel Bus (pohľad zozadu)

Zapojenie portov

- PA1...vstup dát z LA
 PB1...nevyužitý
 PC1...spúšťač slovo
 PA2...bit 7 - RESET - inicializácia LA
 PB2... bit 0, 1, 2 - vzorkovacia frekvencia
 4 - Q1 (0...akt.úroveň H, 1...akt. L)
 5 - Q2

6 - TRUE/FALSE

7 - F in (0...akt.hrana + , 1...akt. -)

PC2.... bit 0 - „programové hodiny“

1 - EXT/INT

2 - R/W

3 - SYN/ASYN

4 - KON

Doska analyzátoru (obr. 3)

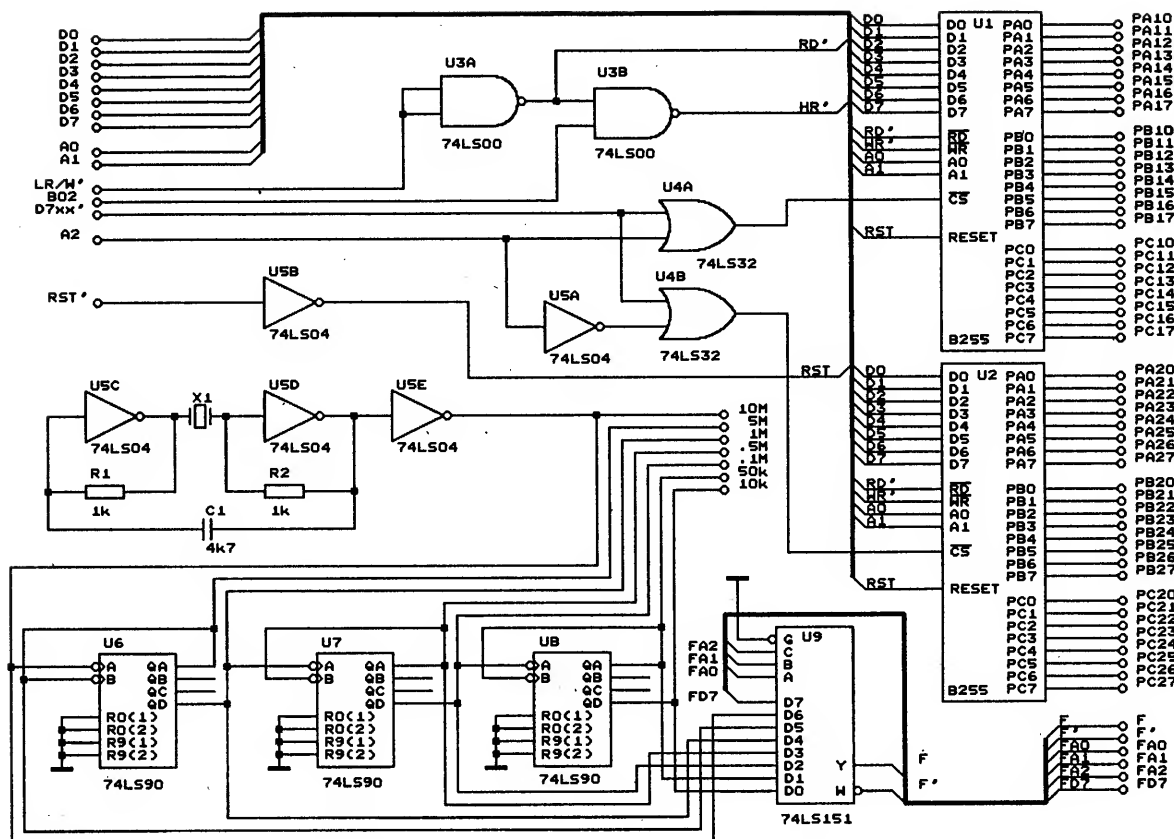
Obsahuje obvody pre spustenie vzorkovania, adresový čítač, pamäť a riadiacu logiku. Podrobný popis bol uvedený v spomenutom článku, rozdiely sú minimálne - upravený je obvod interného spustenia.

Podmienkou spustenia je vznik signálu ZHODA (U10D/13). Brána PC1 určuje spúšťač slovo, takže na výstupoch komparátorov U2, U3 je v prípade rôznych signálov log. 1. Brána PB1 určuje na ktorých bitoch vstupného slova záleží. Ak je príslušný bit PB1 nastavený na log. 1, zodpovedajúci tranzistor zostane zatvorený aj vtedy, keď je na výstupe komparátora log. 1.

Obvody U7-U9 by bolo asi vhodnejšie nahradiť synchronnými čítačmi 74193, pretože asynchrónny čítač má väčšie oneskorenie signálu zo vstupu A na výstup QD, čo sa môže nepriaznivo prejaviť pri vyšších vzorkovacích frekvenciách (pre $f = 10$ MHz trvá celý cyklus zápisu do pamäte 100 ns).

Ako pamäť som vyskúšal obvody MH 82S11, ale pre veľkú spotrebu som ich nahradil jedným obvodom TMM 2016-10 (RAM 2048x8 bitov) s prístupovou dobou 100 ns, ktorého spotreba je výrazne nižšia.

Návrh plošného spoja neužívam, pretože závisí od použitých súčiastok. Problémy pri stavbe by sa mohli vyskytnúť so zápisom do obvodov PPI, pretože časovanie zbernice μ P 6502 je rozdielne oproti zbernici 18080, ale v praxi sa toto zapojenie osvedčilo a sám som ho úspešne vyskúšal s viacerými obvodmi PPI.



Obr. 1. Paralelný port pre Atari 800 XL/XE

Programové vybavenie

Program pre logický analyzátor umožňuje nastavenie parametrov pre snímanie signálov, riadenie činností analyzátoru a prezeranie zosnímaných priebehov.

Obrazovka je rozdelená na tri časti: vrchná - príkazová, stredná - obrazová a spodná - informačná. V príkazovej časti sú zobrazované príkazy vo forme okien, ktoré sa vyberajú pomocou kláves, vymenovaných v informačnej časti:

	pohyb kurzora o príkaz hore, dole,
SHIFT+	skok na prvý, posledný príkaz,
RETURN	vykoná aktuálny príkaz a skočí do nasledujúceho okna,
TAB	zruší aktuálne okno a vráti sa do predošlého,
CTRL+TAB	zruší všetky okná a vráti sa do základného (MODE).

Iba pre okno WORD:

- -	pohyb kurzorom v riadku, kurzor na začiatok, koniec riadku,
SHIFT+	nastavenie slova pre interné spúšťanie, signál ľubovoľnej úrovne (dont care),
0, 1	

pri vysvietenom nápisu Start:

Q preruší sa čakanie na signál KON od analyzátoru.

Ďalšími klávesami sa môže horizontálne pohybovať stredná časť obrazovky alebo značka:

A,S	pohyb obrazovým oknom doľava, doprava,
SHIFT+A,S	rýchly pohyb okna,
Z,X	pohyb značkou doľava, doprava,
SHIFT+Z,X	rýchly pohyb značky,
SPACE	zapamätá si súčasnú polohu značky (pre výpočet vzdialenosti).

Pretože v pamäti logického analyzátoru je 1024 vzoriek a obraz je „šírký“ iba 320 bodov, musí byť zabezpečené prezeranie celého priebehu roiovania obrazu. Na to sú využité možnosti videoprocesora ANTIC, zabudovaného v počítači. Hardwarový register HSCROL (\$D404) umožňuje jemný posun obrazu o 2 pixely, ale iba v rozsahu 16-ich pixelov oboma smermi. Ak je potrebný väčší posun, dosahuje sa to prepisovaním adries v zobrazovacom programe ANTIC-u (Display List). Všetky tieto zmeny sa musia uskutočniť počas trvania vertikálneho zatemňovacieho impulzu, aby sa obraz nerozblkal. V programe je to zabezpečené rutinou VBI, ktorá je vyvolaná

vždy v momente, keď sa lúč vracia do ľavého horného rohu - každých 20 ms. Pretože dĺžka programu VBI je obmedzená na asi 20 000 strojových cyklov, musí byť program VBI rozdelený na dve časti. Tieto sú vyvolávané striedavo - to zabezpečuje premenná DF.

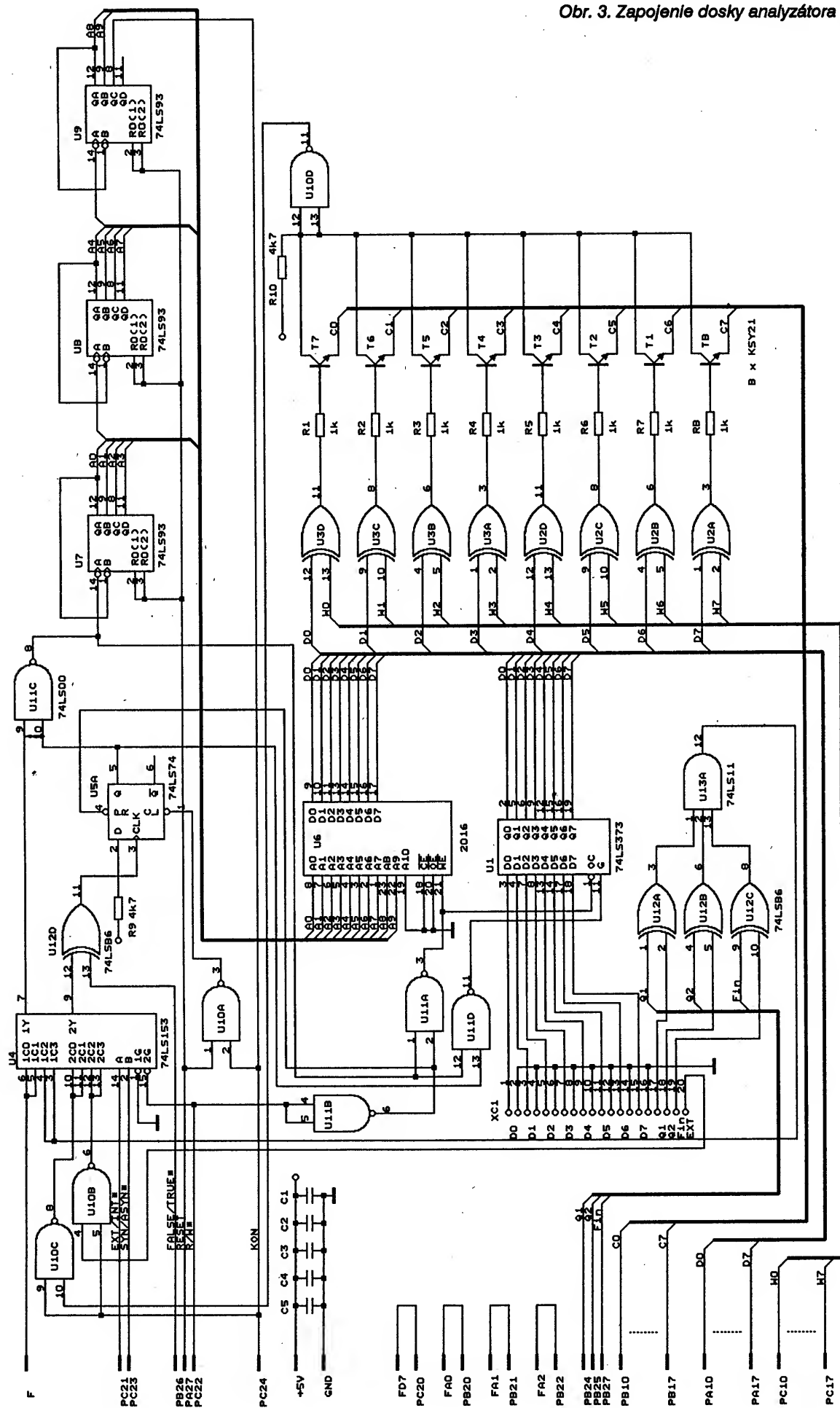
V jednej časti sa vykonávajú všetky posuvy obrazu alebo značky, v druhej sa počíta a zobrazuje okamžitá vzdialenosť značiek.

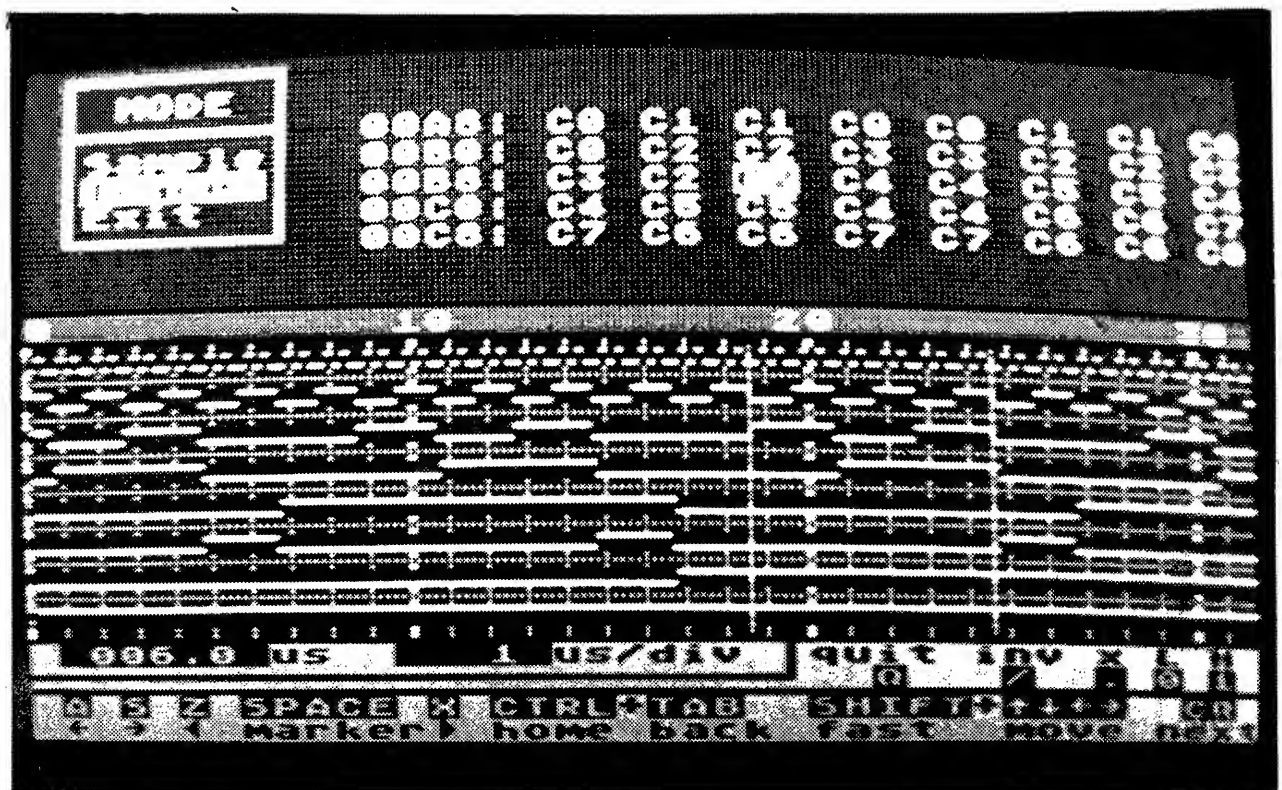
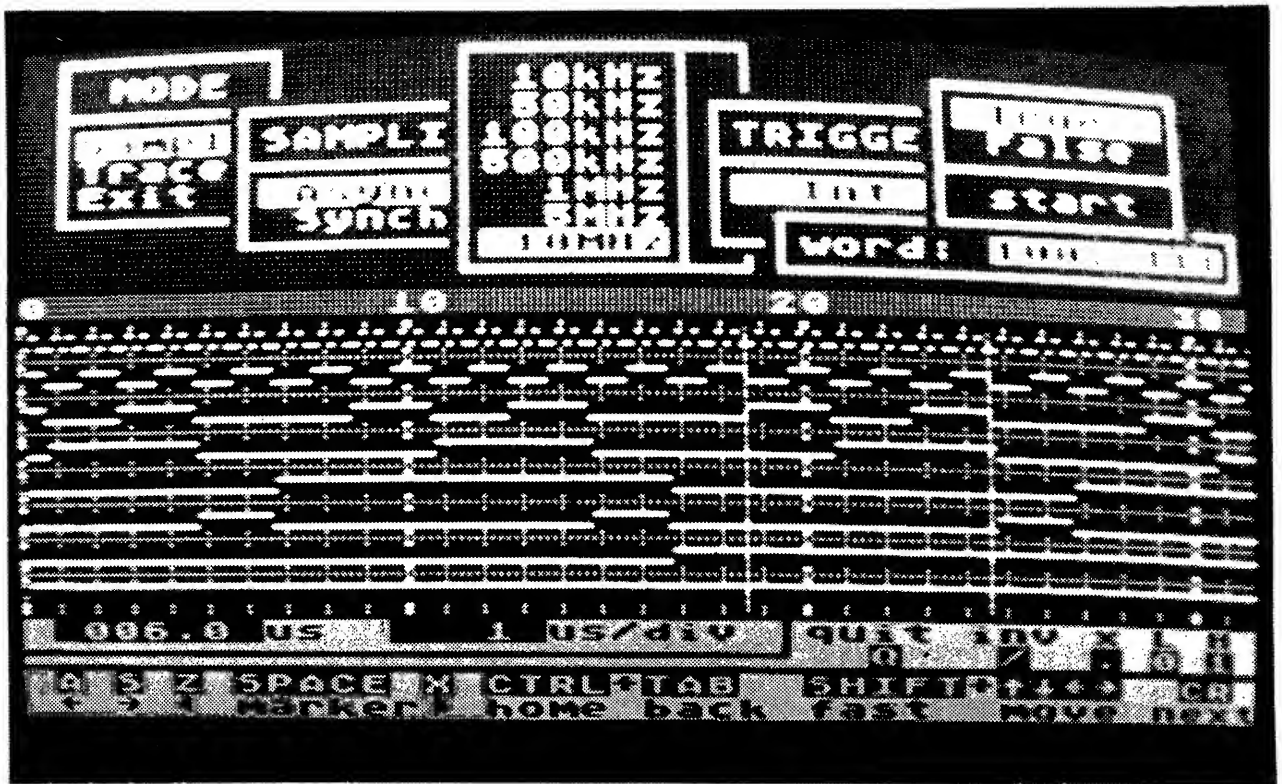
Ďalšia rutina pre obsluhu prerušenia - DLI (Display List Interrupt) umožňuje použiť viac farieb na obrazovke naraz. Táto rutina je vyvolaná vždy, keď ANTIC narazí v DL na zobrazovaciu inštrukciu s najvyšším bitom nastaveným na 1. Vtedy môže CPU zmeniť obsah farbových registrov, takže nasledujúci riadok bude zobrazený v iných farbách.

Celý program je napísaný v strojovom kóde a je umiestnený od adresy \$1200 po \$2280 (dĺžka asi 4kB). Štartuje sa od adresy MAIN. Ak je nutné, aby pracoval v inej pamäťovej oblasti, treba vhodne rozmiesť videopamäte jednotlivých kanálov. Nesmú byť uložené na rozhraní 4 kB blokov, pretože vtedy dochádza k popleteniu ANTIC-u (ANTIC má len 12 bitový adresový čítač).

Výpis programu není vzhľadom k jeho rozsahu možné otisknúť; kto má o program zájem, môže si o ňj napísať autorovi článok - pozn.red.

Obr. 3. Zapojenie dosky analyzátora





Obr. 4. Ukázky práce logického analyzátoru z ATARI 800 XE/XL